МИКРОПРОЦЕССОРЫ

0 1
1 0

СПРАВОЧНИК



МИКРОПРОЦЕССОРЫ

Справочное пособие
для разработчиков
судовой РЭА

Под редакцией канд. техн. наук
Ю. А. Овечкина

Издание 2-е, стереотипное



Ленинград
«Судостроение»
1988

ББК 32.844
М59
УДК 681.325-181.4 (083.72)

Авторы:
Г. Г. ГРИШИН, А. А. МОШКОВ, О. В. ОЛЬШАНСКИЙ, Ю. А. ОВЕЧКИН

Рецензенты: канд. техн. наук, доцент **Ю. М. Щеглов**, инж. Г. С. Алексеев

Микропроцессоры: Справочное пособие для разработчи-
М59 ков судовой РЭА/Г. Г. Гришин, А. А. Мошков, О. В. Ольшанский, Ю. А. Овечкин. — 2-е изд., стереотип. — Л.: Судостроение, 1988. — 520 с., ил.
ISBN 5-7355-0306-5

В книге приведены справочно-информационные материалы по отечественным микропроцессорам. Изложены общие принципы построения аппаратуры на базе микропроцессоров. Даны конкретные рекомендации по применению и выбору типа микропроцессора в судовой радиоэлектронной аппаратуре (РЭА).
Для разработчиков судовой РЭА.

М $\frac{3605030000-C60}{048(01)-88}$ КБ-14-37-88 ББК 32.844

Справочное издание

Гришин Георгий Георгиевич
Мошков Алексей Алексеевич
Ольшанский Олег Владимирович
Овечкин Юрий Алексеевич

МИКРОПРОЦЕССОРЫ

Справочное пособие
для разработчиков
судовой РЭА

Заведующий редакцией *П. К. Зубарев*. Редактор *А. И. Голикова*
Художественный редактор *О. П. Андреев*
Технический редактор *Р. К. Чистякова*
Корректоры: *Т. С. Александрова*, *С. Н. Маковская*, *В. Ю. Самохина*
Оформление художника *Б. Н. Осенчакова*

ИБ № 1544

Подписано в печать 27.07.88. М-27704. Формат 60×90 1/16.
Бумага типографская № 2. Гарнитура литературная. Печать высокая. Усл. печ. л. 32,5.
Усл. кр.-отт. 32,5. Уч.-изд. л. 36,4. Тираж 33 500 экз. Изд. № 4346—88.
Заказ № 611. Цена 2 р. 10 к.

Издательство «Судостроение», 191065, Ленинград, ул. Гоголя, 8

Отпечатано с матриц Ленинградской типографией № 6 ордена Трудового Красного Знамени Ленинградского объединения «Техническая книга» им. Евгении Соколовой Союзполиграфпрома при Государственном комитете СССР по делам издательств, полиграфии и книжной торговли.
193144, г. Ленинград, ул. Моисеенко, 10.

ISBN 5-7355-0306-5 © Издательство «Судостроение», 1987

ПРЕДИСЛОВИЕ

Качественный скачок в области производства полупроводниковых приборов, в частности технологии их изготовления, позволил создать принципиально новый тип элементной базы — большую интегральную схему (БИС) с размещенными в одном кристалле десятками и даже сотнями тысяч электронных элементов, скоммутированных между собой по определенной схеме. Такое решение привело к резкому повышению плотности компоновки электронных элементов, но одновременно с этим к определенному снижению универсальности отдельных микросхем. Стремление ликвидировать этот недостаток, создать универсальные БИС широкого применения и привело к изобретению микропроцессора— программируемой БИС. Создание микропроцессоров можно считать одним из важнейших достижений двух быстроразвивающихся отраслей промышленности: вычислительной техники и микроэлектроники.

Сегодня в элементной базе, включающей большие и сверхбольшие интегральные схемы (БИС, СБИС), однокристальные микроЭВМ, нескоммутированные логические матрицы и другое, микропроцессоры занимают ведущее место. Особое значение придается расширению их логических возможностей и функциональной насыщенности.

На современном этапе развития судового приборостроения одним из основных компонентов, оказывающих существенное влияние на тактико-технические характеристики радиоэлектронного оборудования судов, является применяемая элементная база. Правильный выбор элементной базы на 80—90 % определяет технико-экономические характеристики судовой аппаратуры. Применение микропроцессорных комплектов (МПК) БИС в судовой РЭА повышает ее надежность и производительность, уменьшает массогабаритные характеристики и потребляемую мощность.

Показатель надежности является одной из самых важных характеристик судовой системы. Повышение надежности аппаратуры при использовании в качестве элементной базы МПК можно обеспечить:

резким сокращением числа электронных схем, необходимых для реализации требуемых вычислительных и логических функций и, как следствие, уменьшение связей между ними;

созданием резервированных вычислительных систем, что достигается малыми габаритами и большими функциональными возможностями микропроцессорной техники;

построением распределенных систем, обладающих большей живучестью и надежностью.

Сокращение объема информации, передаваемой между отдельными разнесенными частями системы, также способствует повышению надежности функционирования системы. В частности, в распределенных системах, когда вычислительные средства максимально приближены к источникам информации и объектам управления, снижены требования к широкополостности каналов связи. Построение распределенных систем на МПК решает и такую задачу, как увеличение производительности вычислительных средств. При этом предполагается, что алгоритм позволяет производить параллельную обработку информации на нескольких вычислителях.

Одним из преимуществ микропроцессорных систем является их гибкость, так как логика их функционирования определяется программой, хранимой в ПЗУ или ОЗУ. В результате возможно значительное изменение характеристик системы только за счет замены программы в памяти.

Применение МПК в качестве элементной базы позволяет с успехом решать такую важную задачу, как уменьшение стоимости разработки аппаратуры и ее серийного производства. Это обеспечивается переходом от кропотливого и дорогостоящего конструирования аппаратуры на логических элементах малой степени интеграции к ее проектированию на основе крупных функционально законченных узлов (МПК БИС).

В настоящее время микропроцессоры находят самое широкое применение в судовой аппаратуре. Это личные компьютеры экипажа, микроЭВМ, встроенные в исполнительные механизмы и датчики, аналоговые микропроцессорные устройства, большие децентрализованные управляющие и информационные системы и многое другое.

Трудно найти судовую аппаратуру, связанную с получением, передачей, обработкой и отображением информации, в которой применение микропроцессоров не приводило бы к улучшению технико-экономических показателей.

В ряде случаев открываются принципиально новые возможности построения систем. Так, например, применение микропроцессоров в судовых автоматизированных управляющих системах изменяет стратегию управления и делает целесообразным переход от централизованных систем управления к децентрализованным или распределенным, в которых не только рассредоточена аппаратура, но и осуществлено распределение функций между отдельными микропроцессорными системами.

Особенно эффективно применение микропроцессоров в тех судовых узлах, где ранее невозможно было использование вычисли-

тельной техники из-за больших размеров или недостаточной надежности.

Однако широкое внедрение МПК БИС зависит от умения разработчиков судовой РЭА правильно применять эту новую элементную базу, что требует от них знания как схемотехники и логической структуры БИС, так и особенностей построения аппаратуры на их базе.

В СССР издано значительное количество работ по микропроцессорам. Это издания справочного характера, в которых рассмотрены несколько МПК БИС. Однако описания в них носят иллюстративный характер и дают лишь общую информацию [2, 34, 45]. Выпущены книги, в которых подробно рассмотрены отдельные БИС МПК и особенности их применения [56]. В этих работах, как правило, отсутствуют статические и динамические параметры.

В настоящем справочном пособии сделана попытка восполнить этот пробел, т. е. дать не только описание функционирования и структуры БИС МПК, но и привести статические и динамические параметры шести наиболее перспективных для судовой РЭА МПК БИС, особенности проектирования РЭА, выбор оптимальной серии БИС и конструктивно-технологические особенности их применения.

Справочное пособие не заменяет официальных документов (технических условий, указаний по применению, паспортов на микросхемы, другой нормативно-технической документации), и приведенные параметры микросхем могут использоваться для предварительных расчетов, которые должны уточняться по соответствующим частным техническим условиям.

Гл. 1 вводная, в ней приведены краткие сведения по архитектуре, классификации, технологии изготовления микропроцессоров. В гл. 2 показаны области применения микропроцессоров в судовой РЭА, описаны примеры судовой аппаратуры на их базе. Гл. 3—8 посвящены МПК БИС различных серий и могут изучаться независимо друг от друга. В гл. 9 даны рекомендации по применению микропроцессоров. Эту главу желательно прочитать всем разработчикам независимо от того, какой МПК БИС их интересует. В гл. 10 приведены классификация и описания систем проектирования микропроцессорной РЭА.

Для однозначного понимания излагаемого материала в конце книги предложен список терминов, наиболее часто встречающихся в работе. Определения терминов заимствованы в основном из нормативных документов [11, 15—19].

Гл. 1 и 2 написаны А. А. Мошковым; гл. 3—8 — Г. Г. Гришиным, Ю. А. Овечкиным, О. В. Ольшанским; гл. 9 — О. В. Ольшанским; гл. 10 — Г. Г. Гришиным, О. В. Ольшанским.

Отзывы и предложения просим присылать по адресу: 191065, Ленинград, ул. Гоголя, д. 8, издательство «Судостроение».

СПИСОК ОСНОВНЫХ СОКРАЩЕНИЙ И ОБОЗНАЧЕНИЙ

АЗУ — ассоциативное запоминающее устройство
АЛУ — арифметико-логическое устройство
АР — арифметический расширитель
АСУ — автоматизированная система управления
АУ — арифметическое устройство
БИС — большая интегральная схема
БМУ — блок микропрограммного управления
ВС — вычислительная система
ДБК — дуга большого круга
ЗУ — запоминающее устройство
ИС — интегральная схема
И²Л — интегральная инжекционная логика
МИС — малая интегральная схема
МП — микропроцессор
МПК — микропроцессорный комплект
ОБ — операционный блок
ОЗУ — оперативное запоминающее устройство
ПЗУ — постоянное запоминающее устройство
ППЗУ — программируемое постоянное запоминающее устройство
РК — регистр команд
РМК — регистр микрокоманд
РОН — регистр общего назначения
РЭА — радиоэлектронная аппаратура
СБИС — сверхбольшая интегральная схема
СИС — средняя интегральная схема
СК — счетчик команд
СОЗУ — сверхоперативное запоминающее устройство
ТТЛ — транзисторно-транзисторная логика
ТТЛШ — транзисторно-транзисторная логика с диодами Шоттки
УБ — управляющий блок
УВМ — управляющая вычислительная машина
УП — управляющая память
ЦВК — цифровой вычислительный комплекс
ЦВМ — цифровая вычислительная машина
ЦВС — цифровая вычислительная система
ЦП — центральный процессор
ЭВМ — электронная вычислительная машина
ЭППЗУ — электрически программируемое постоянное запоминающее устройство
ЭСЛ — эмиттерно-связанная логика
X — уровень сигнала безразличен
«1» — логическая единица
«0» — логический нуль

Глава 1

МИКРОПРОЦЕССОРЫ. ОБЩИЕ СВЕДЕНИЯ

1.1. АРХИТЕКТУРНЫЕ ОСОБЕННОСТИ И КЛАССИФИКАЦИЯ МИКРОПРОЦЕССОРОВ

Микропроцессор (МП) по своим функциям эквивалентен процессору ЭВМ. В общем случае МП позволяет выполнять следующие операции: прием закодированных инструкций; прием, обработку, хранение и вывод закодированной информации; ввод и вывод сигналов, управляющих работой микропроцессорных БИС и других схем или характеризующих их состояние.

В настоящее время в мире выпускают более ста типов микропроцессоров. Все они различаются, однако любой из них предназначен для выполнения последовательности команд (микрокоманд), предусмотренных программой решения задачи.

Такое многообразие МП определяется различным сочетанием их характеристик, некоторое представление о котором можно получить из приведенной на рис. 1.1 классификации МП.

Особенность МП как устройства с программируемой логикой заключается в подчиненности его структуры программируемости [4]. Это означает, что функции, реализуемые МП, определяются не столько его структурой, сколько последовательностью управляющих слов (команд), поступающих из программной памяти на входы МП. При изменении этой последовательности изменяется и выполняемая МП функция. Поэтому разработчик судовой РЭА при анализе функциональных возможностей МП должен учитывать не только его структуру, но и возможности программной реализации функций. Для комплексной характеристики возможностей МП будем пользоваться термином «архитектура».

Архитектура МП — это его логическая организация, определяемая возможностями МП по аппаратной или программной реализации функций, необходимых для построения микроЭВМ. Понятие архитектуры МП отражает его структуру, способы обращения ко всем доступным для пользователя элементам структуры, способы представления и форматы данных, набор операций, выполняемых МП, способы указания (адресации) данных, участвующих в операциях, форматы управляющих слов, поступающих в МП извне, характеристики и назначение вырабатываемых МП управляющих сигналов, реакцию МП на внешние сигналы.

Микропроцессор, как и любое устройство обработки цифровой информации, можно разделить на операционный и управляющий блоки [13]. Такой подход упрощает проектирование, а также

облегчает понимание функционирования устройства [35] (рис. 1.2).

Операционный блок (ОБ) состоит из регистров, сумматоров и других узлов, производящих прием из внешней среды и хранение кодов слов, их преобразование и выдачу во внешнюю среду

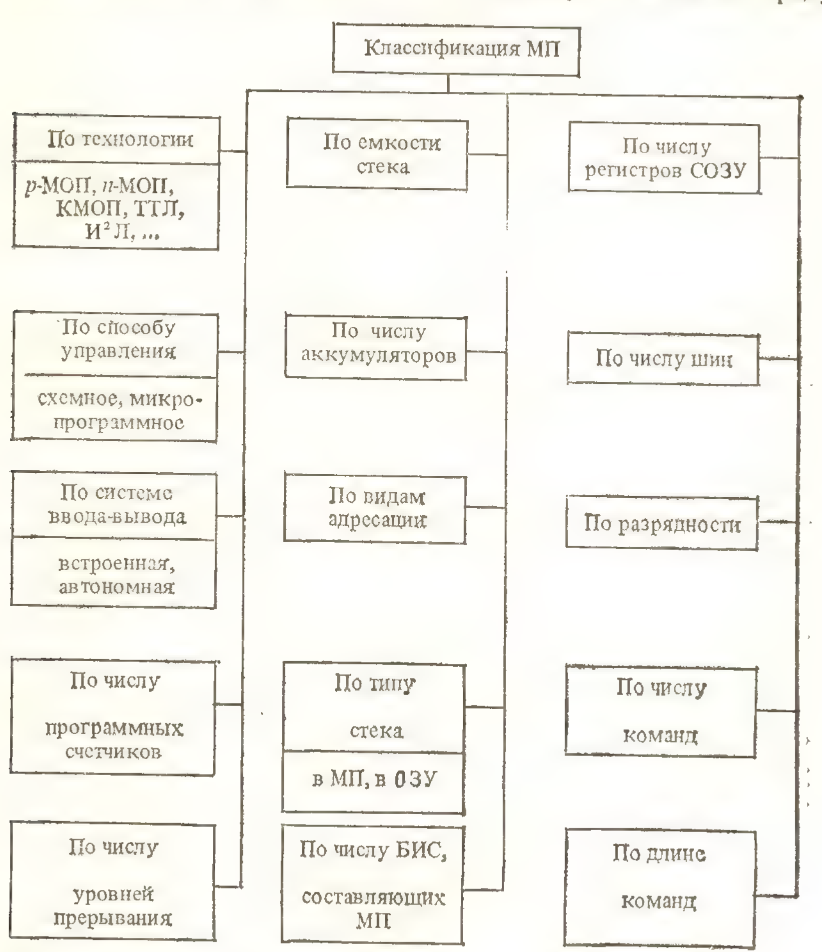


Рис. 1.1. Классификация микропроцессоров

результата обработки, а также выдачу в *управляющий блок* и внешнюю среду *оповещательных сигналов* о знаках и особых значениях операндов, их отдельных разрядов, особых значениях промежуточных и конечных результатов операции (например, равенства нулю результата операции и др.).

Процесс функционирования во времени устройства обработки цифровой информации состоит из последовательности тактовых

интервалов, в которых ОБ производит определенные элементарные операции преобразования слов. ОБ выполняет некоторый набор элементарных преобразований информации, например таких, как передача слова из одного регистра в другой, взятие обратного кода, сдвиг и др. Выполнение этих элементарных операций инициируется поступлением в ОБ соответствующих *управляющих сигналов*.

Элементарная функциональная операция (или их некоторая комбинация), выполняемая за один тактовый интервал и приводимая в действие одним управляющим сигналом, называется *микрооперацией*.

В некоторые такты могут поступать несколько управляющих сигналов, вызывая параллельные во времени выполнения соответствующих микроопераций.

Управляющий блок (УБ) вырабатывает распределенную во времени последовательность управляющих сигналов, порождающих в ОБ нужную последовательность микроопераций.



Рис. 1.2. Декомпозиция микропроцессора

Последовательность управляющих сигналов определяется сигналами *кода операции*, поступающими в УБ извне и *оповещательными сигналами*, зависящими от *операндов* и промежуточных результатов преобразований.

Операционный блок задается его структурой, т. е. составом узлов и связями между ними, и выполняемым им набором микроопераций.

Основой операционного блока МП является комбинационное арифметико-логическое устройство (АЛУ). АЛУ служит для выполнения арифметических и логических преобразований над словами, называемыми в этом случае *операндами*.

В процессе выполнения операций АЛУ взаимодействует с регистрами МП, являющимися обычно источниками и приемниками операндов для такого АЛУ, при этом, как правило, один и тот же регистр может рассматриваться и как источник, и как приемник информации. Для реализации такой возможности необходимо осуществлять временное запоминание промежуточных результатов на отдельных регистрах. С этой целью используют либо регистры для кратковременного запоминания операндов, либо регистры для кратковременного запоминания результата.

Другим важным узлом ОБ МП является набор регистров общего назначения (РОН), часто называемый СОЗУ. В РОН хра-

нятся операнды и результаты обработки информации в АЛУ. Обращение к РОН — адресное. Эти регистры допускают считывание и запись информации, поэтому содержат входную и выходную шины, адресную шину и управляющий вход, информация на котором задает режим работы: запись, хранение или чтение информации.

Рассмотрим типовую структуру ОБ МП (рис. 1.3). При этом не будем указывать управляющие входы, что позволит основное внимание сосредоточить на функциональных характеристиках

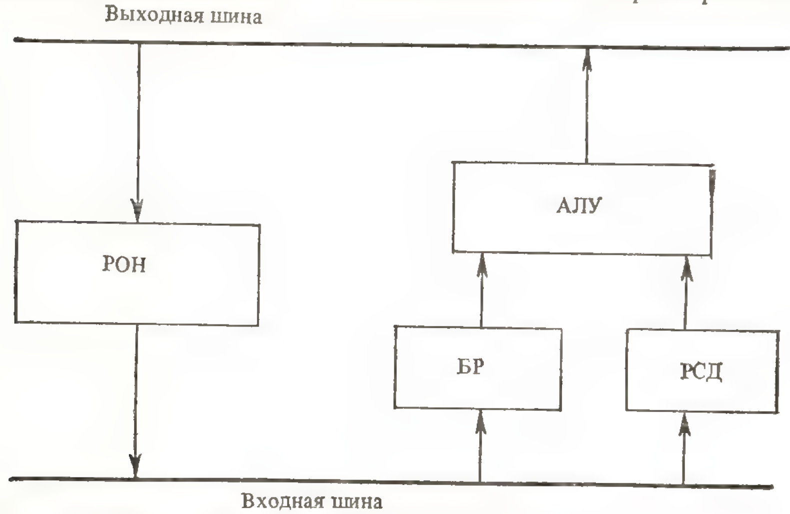


Рис. 1.3. Типовая структурная схема операционного блока микропроцессора

системы при обработке данных. Представленная структура обладает очень широкими возможностями. Содержимое любого *РОН* может быть передано на буферный регистр *БР* и на регистр сдвига *РСД*. *АЛУ* может выполнять логические и арифметические операции над содержимым обоих регистров, результат может быть записан в любой *РОН*.

В схеме должна быть предусмотрена возможность анализа требуемых логических условий. В общем случае число этих условий может быть достаточно велико, что облегчает составление программ решения задач. Для упрощения структуры МП и организации его работы признаки условий обычно хранятся на специально выделенном регистре — регистре признаков (РП), разрядность которого не превышает разрядности внутренних шин МП. При этом легко организуется засылка и временное хранение содержимого этого регистра (слово состояния) например, при переходе к подпрограммам.

Если в схему (см. рис. 1.3) добавить цепь переноса (*АЛУ*) с хранением сигнала переноса и цепь сдвига (*РСД*) с хранением

значения вытесняемого при сдвиге разряда, то становятся возможными операции над словами с разрядностью, больше, чем разрядность *АЛУ*, *РОН* и вспомогательных регистров. Учитывая также то, что на *входную шину* можно подавать информацию извне, а с *выходной* снимать эту информацию, то рассмотренная структура позволяет выполнять любую логическую и арифметическую обработку информации под воздействием управляющих сигналов, поступающих из управляющего блока (устройства управления).

Рассмотрим вопросы организации управляющего блока. Существуют два основных метода построения логики УБ.

Управляющий блок с жесткой или схемной логикой. Для каждой операции, задаваемой, например, кодом операции команды, строится набор комбинационных схем, которые в нужных тактах вырабатывают соответствующие управляющие сигналы. Иначе говоря, строится конечный автомат, в котором необходимое множество состояний релизуется на запоминающих элементах, а функции переходов и выходов реализуются с помощью комбинационных схем.

Управляющий блок с хранимой в памяти логикой (*с микропрограммным управлением*). Любой выполняемой в ОБ операции соответствует совокупность хранимых в памяти слов — микрокоманд, каждая из которых содержит информацию о микрооперациях, подлежащих выполнению в течение одного машинного такта, и указание (в общем случае зависящее от значения входных сигналов), какое слово должно быть выбрано из памяти следующим (следующая микрокоманда). Таким образом, в этом случае функции переходов и выходов реализуются хранимой в памяти совокупностью микрокоманд.

Последовательность микрокоманд, выполняющих одну машинную команду или отдельную процедуру, образует микропрограмму. Обычно микропрограммы хранятся в специальной памяти микропрограмм (управляющей памяти). Метод проектирования и реализации микропрограммного УБ с использованием управляющей памяти (УП) получил название *микропрограммирования.*

Хранение микрокоманд в УП позволяет легко вносить изменения в систему команд МП: для введения новой операции достаточно записать в УП микропрограмму ее выполнения. Это одно из достоинств микропрограммного способа управления. Однако необходимость в каждом такте обращаться к УП ограничивает максимальную частоту работы УБ быстродействием УП.

Как и в обычных ЭВМ последовательность команд (программа), указывающая на нужный порядок выполнения операций, размещается во внешнем по отношению к МП ЗУ. Таким образом, МП должен обеспечивать выборку команд в нужной последовательности, их дешифрацию, выполнение некоторых действий в соответствии с содержанием полей команды и передачу кода команды в УБ.

Для выполнения этих функций в МП должны быть предусмотрены специальные средства: счетчик команд (СК), предназначенный для хранения текущего адреса команды; регистр команд (РК), предназначенный для приема и хранения поступающих из ЗУ команд; схемы выдачи адресов операндов и содержимого СК на адресную шину МП; схемы приема данных и команд с внешней шины данных, дешифратор команд (ДШК).

Рассмотренная архитектура МП (см. рис. 1.3) с двумя шинами не является единственной. В различных МП используются одна, две или три внутренних шины; их число существенно влияет на структуру и характеристики МП.

В МП обычно предусматривают возможность работы с магазинной памятью — стеком, при обращении к которой не требуется указания адреса. В МП используют два вида стека: встроенный и автономный. Встроенный стек размещают полностью на кристалле МП и его емкость (глубина стека) не может быть очень большой. Отличие автономного стека от встроенного в том, что в нем в качестве накопителя используется внешнее ОЗУ. Непосредственно на кристалле располагают лишь указатель стека, с разрядностью, равной разрядности шины адреса. Использование ОЗУ значительно увеличивает глубину стека и время обращения к стеку. Для компенсации снижения быстродействия в некоторых МП с автономным стеком реализованы аппаратная запись и восстановление при прерываниях содержимого РК и других узлов.

Структура МП, содержащая все указанные узлы, представлена на рис. 1.4. В МП имеются две *шины данных: шина А* и *шина В*. Адрес подлежащей выполнению команды хранится на *СК*. С *СК* он поступает через буфер адреса *БА*, предназначенный для повышения нагрузочной способности *шины адреса*, на адресные входы ЗУ. Команда, выбранная из ЗУ по сигналу Y_i, поступает через буфер данных *БД* на *РК*. Код команды расшифровывается дешифратором команд *ДШК*, который анализирует отдельные поля команды и передает код операции в устройство управления операциями *УУО*. *УУО* в соответствии с кодом операции вырабатывает последовательность управляющих сигналов Y_i, обеспечивающих выполнение нужной операции. Если в процессе выполнения операции требуется обращение к *РОН*, то *ДШК* выставляет адрес регистра на адресных входах *РОН*. Если выполнение операции связано с обращением к внешнему по отношению к МП ЗУ, то соответствующий адрес поступает на *шину адреса* через *БА*. Следует отметить, что адресное поле команды в МП обычно засылается не в *РК*, а в один из *РОН*. В процессе выполнения операции изменяется содержимое счетчика команд, аккумулятора, регистра состояния и некоторых других узлов.

Рассмотренные выше функциональные и структурные особенности МП позволяют составить обобщенную схему МП (рис. 1.5), основой которой является структурная схема, приведенная на

рис. 1.4. В эту обобщенную схему включены: *стек*, индексные регистры *ИР*, аккумулятор *А*, регистр признаков *РП*, схема инкремента-декремента *СИД*, блок прерываний *БП*.

Конкретные МП, как правило, не содержат всех узлов, показанных на схеме рис. 1.5. В ряде МПК отдельные функциональные узлы выполняются автономно — в виде БИС или СИС.

Различные способы распределения функциональных узлов МП по отдельным БИС позволяют выделить три основных типа МП [4]:

функционально законченный однокристальный с фиксированной разрядностью и фиксированной системой команд;

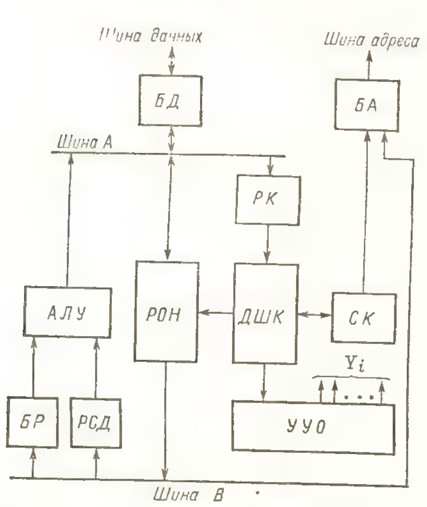


Рис. 1.4. Структурная схема микропроцессора

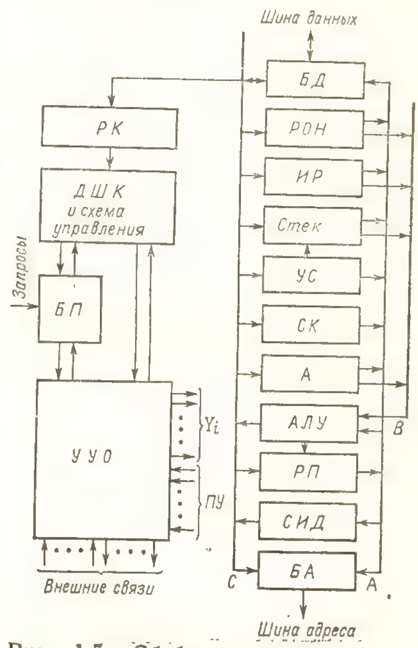


Рис. 1.5. Обобщенная структурная схема микропроцессора

повышенной разрядности на основе одной БИС ОБ и одной или нескольких БИС УБ; границей такого разбиения МП на рис. 1.5 является шина *С*;

произвольной разрядности из нескольких однотипных БИС ОБ и одной или нескольких БИС УБ.

Функциональная законченность и автономность МП первого типа при заданном уровне интеграции обеспечиваются размещением на кристалле операционного и управляющего блоков, схемы прерывания, канала обмена информацией, выходных каскадов согласования внутренних и внешних сигналов.

Характерной особенностью МП второго и третьего типов является раздельная реализация БИС ОБ и БИС УБ. Существенным различием между МП второго и третьего типов является способ организации ОБ.

В МП второго типа ОБ полностью размещается на одном кристалле и для него сохраняются ограничения на число трактов передачи информации. В то же время МП второго типа приобретает преимущества МП с микропрограммным управлением.

В МП третьего типа БИС ОБ представляет собой модуль, предназначенный для обработки 2 ... 16 разрядов информации (хотя может быть и больше). Для объединения таких БИС в единый ОБ заданной разрядности, кратной разрядности одной БИС, необходимо иметь в БИС независимые входы и выходы переноса и сдвига. МП третьего типа называют также секционированными (с наращиваемой разрядностью) с микропрограммным управлением.

1.2. ОСОБЕННОСТИ ТЕХНОЛОГИИ ИЗГОТОВЛЕНИЯ МИКРОПРОЦЕССОРОВ

Развитие технологии изготовления микропроцессоров проходило по известным для интегральных микросхем направлениям: *p*- и *n*-канальные униполярные (*n*-МОП * и *p*-МОП), униполярные с дополнительной симметрией или комплементарные (КМОП) структуры; биполярные структуры на базе транзисторно-транзисторной логики (ТТЛ), эмиттерно-связанной логики (ЭСЛ) и интегральной инжекционной логики ($И^2Л$). С точки зрения технологии, микропроцессоры прошли тот же путь, что и микросхемы малой интеграции. Первые микропроцессоры, выполненные на основе *p*-МОП структур, отличались простой технологией и высокой степенью интеграции, однако имели низкое быстродействие и требовали несколько источников питания.

Создание микропроцессоров на основе *n*-МОП-структур позволило получить более высокое быстродействие. Большинство 8- и 16-разрядных однокристальных микропроцессоров выполнены на *n*-МОП-структурах.

Широкое развитие получили КМОП схемы. Они имеют высокую помехоустойчивость, работают от стандартного (+5 В) источника питания и представляют собой самые микромощные схемы.

Микропроцессоры, выполненные по биполярной технологии, отличаются высоким быстродействием, однако они рассеивают значительную мощность и изготовление их связано с более сложным технологическим процессом, что обусловливает их менее высокую степень интеграции, чем МОП БИС.

Наиболее распространенными, изготовленными по биполярной технологии, являются ТТЛШ-микропроцессоры.

Одним из направлений биполярной технологии являются $И^2Л$-схемы. Их особенностями являются токовый характер питания и малая мощность потребления.

---

* МОП означает металл—окисел—полупроводник (или МДП — металл—диэлектрик—полупроводник)

ЭСЛ-микропроцессоры самые быстродействующие, но обладают и самым высоким потреблением мощности. Этот недостаток сдерживает пока достижение высокой степени интеграции ЭСЛ-микропроцессоров.

Одной из проблем при создании микропроцессоров является достижение максимальной степени интеграции.

Повышение степени интеграции ограничено несколькими факторами: геометрическими размерами элементов, типом активного элемента, размерами кристалла БИС. Увеличение степени интеграции сдерживают также проблемы отвода тепла и тепловые связи между элементами в кристалле.

Что касается геометрических размеров, то ограничением здесь являются, с одной стороны, характеристики элементов, а с другой — возможности технологии и технологического оборудования.

Что касается элементов, то МОП-транзисторы занимают площадь, почти на порядок меньшую, чем биполярные транзисторы. Размеры полупроводникового кристалла ограничиваются дефектами в них: чем больше технологических операций, тем больше вносится в кристалл дефектов и тем ниже процент выхода годных микросхем. Ввиду того, что для изготовления МОП БИС требуется значительно меньше операций, чем для изготовления биполярных БИС при одном и том же проценте выхода годных микросхем, площадь кристалла для них может быть значительно увеличена, а если к тому же учесть, что размер МОП-транзистора более чем на порядок меньше размера биполярного транзистора, то становится понятным, что степень интеграции БИС на МОП-структурах значительно больше, чем на биполярных.

Повышение степени интеграции обеспечивается и использованием новой схемотехники. Одним из направлений является снижение энергетических уровней функционирования. Для этой цели основная часть схем обработки информации микропроцессора строится на базе схем ТТЛШ и ЭСЛ с уменьшенным перепадом логических сигналов.

Выходные же схемы БИС имеют такие же значения логических сигналов, как стандартные микросхемы ТТЛШ или ЭСЛ, для совместной работы БИС с этими микросхемами. Поэтому на входах и выходах БИС должны включаться схемы для преобразования логических уровней.

Для повышения экономической эффективности изготовления БИС в ряде случаев их проектируют на основе базового кристалла, содержащего матрицу однородных логических элементов, из которых, нанося соответствующий рисунок металлических соединений, можно получить БИС с требуемой функцией.

Однородность соблюдается и в структуре логических элементов, т. е. разработчики стремятся строить логические схемы из однородных элементов, так как при этом упрощается топология схемы и уменьшается площадь, занимаемая схемой на кристалле.

Элементная однородность достигается, например, в БИС на МОП-транзисторах.

В большинстве логических схем на биполярных структурах кроме транзисторов используют диоды и резисторы. Для этих схем элементная однородность заключается в том, чтобы обеспечить приблизительно одинаковые площади, занимаемые различными элементами на кристалле.

Чтобы обеспечить схемную однородность БИС, в качестве базовых логических элементов используются схемы, выполняющие функции И—НЕ, ИЛИ—НЕ, либо комбинированные функции И—ИЛИ—НЕ.

Среди логических элементов БИС, изготавляемых по биполярной технологии, интерес представляют схемы $И^2Л$. Вентиль типа $И^2Л$ занимает площадь, равную площади одного биполярного транзистора.

Биполярные микропроцессоры. Для производства микропроцессоров, как правило, применяют планарную технологию, позволяющую на одной пластине кремния получать одновременно несколько десятков или сотен БИС в едином технологическом процессе. При этом создают различные структуры, образующие законченную схему, включающую активные и пассивные элементы. Основные процессы этой технологии те же, что и при производстве микросхем малой интеграции: локальная диффузия, эпитаксиальное выращивание, напыление с чередующимися фотолитографией и окислением.

Основным полупроводниковым материалом является кремний. Он легко поддается селективной диффузии, имеет высокое сопротивление и позволяет получить достаточно высокие рабочие температуры микросхем. На поверхности кремния легко создается окисная пленка, которая служит защитным покрытием при проведении ряда технологических операций и предохраняет готовую схему от внешних воздействий.

Для изготовления микросхем наиболее удобными оказались пленки кремния, полученные методом эпитаксиального выращивания (планарно-эпитаксиальный процесс).

Рассмотрим кратко процесс изготовления полупроводниковой микросхемы (кристалла).

После окисления поверхности пластины необходимо выделить на ней локальные области, в которые должна проводиться диффузия. Для этой цели применяют метод фотолитографии. Для изготовления микросхем нужно несколько разных фотошаблонов (5 ... 20). В окна, образованные в SiO_2, проводят локальную диффузию примесей в кремний для создания диодных или транзисторных структур, резисторов, изолирующих переходов и т. п. При этом окисная пленка предохраняет кремний от нежелательного внедрения примеси на отдельных участках поверхности. Диффузия, как правило, проводится из газовой фазы. В качестве диффундирующих примесей обычно используют бор, фосфор,

сурьму, мышьяк. Глубина диффузии и поверхностная конфигурация диффузионного слоя определяются временем и температурой диффузии.

На рис. 1.6 показан разрез кристалла фрагмента интегральной схемы. Транзисторы размещены в одной изолированной области *1*, а резисторы — в другой *2*. Затемненные участки на поверхности кристалла — металлизация, контактные площадки и межсоединения. Транзисторы имеют общий коллектор. Характеристики микросхем зависят не только от структуры, т. е. распределения по глубине кристалла локальных областей с различной проводимостью, но и от топологии (топологического чертежа) или конфигурации, размеров и формы элементов их взаимного расположения и рисунка межсоединений.

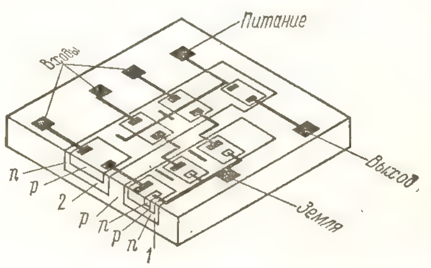


Рис. 1.6. Поперечное сечение фрагмента микросхемы

При переводе электрической схемы в топологическую нужно не просто составить топологию каждого элемента и их соединения, но необходимо учитывать взаимосвязь элементов, обусловленную их взаимодействием и паразитными эффектами, присущими полупроводниковым микросхемам. Основными требованиями при разработке топологии являются: во-первых, достижение минимальных габаритов элементов и возможно высокой плотности их размещения и, во-вторых, изготовление всех элементов в едином технологическом цикле.

Примером совершенствования технологических приемов изготовления БИС может служить микросхема, изготовленная с использованием окисной изоляции.

Разновидность технологических процессов связана в основном с методами изоляции между элементами. Наиболее простым в изготовлении и широко распространенным является метод изоляции посредством обратно смещенного *p*—*n*-перехода. Однако изолирующий переход существенно увеличивает площадь элементов. Более прогрессивной является изопланарная технология (изопланар I). Изоляция элементов в этом случае осуществляется травлением канавок между элементами и последующим окислением поверхности канавок. При этом площадь, занимаемая элементом (с учетом канавки), значительно сокращается, снижаются паразитные емкости, повышается пробивное напряжение.

Еще большее сокращение площади элемента достигается при изоляции между элементами поликристаллического кремния (изопланар II). Эта технология позволяет почти в 3 раза повысить плотность размещения элементов по сравнению с традиционной планарной технологией, а также снизить паразитную емкость.

Дальнейшее сокращение размеров достигается применением анизотропного травления канавок, т. е. травлением вдоль кристаллографических осей.

На рис. 1.7 показана геометрия обычного транзистора, а также транзисторов, выполненных по технологии изопланар I и изопланар II. На рис. 1.7 видно, что площадь, занимаемая транзистором, выполненным по технологии изопланар II, почти в три раза меньше площади обычного транзистора. Это способствует повышению степени интеграции БИС и увеличению их быстродействия.

Одна из наиболее сложных задач — соединить большое число элементов в одном кристалле. Обычно соединения в БИС выполняются по многоуровневой системе. Первый уровень — это связи, объединяющие отдельные элементы в простейшие логические схемы И-НЕ, ИЛИ-НЕ, триггер и т. д.

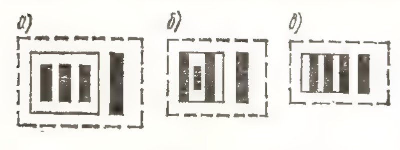


Рис. 1.7. Геометрия транзисторов, выполненных по трем видам технологии: *а*) обычная; *б*) изопланар I; *в*) изопланар II

Второй уровень — объединение элементарных логических схем в регистры, счетчики, сумматоры, дешифраторы и т. д. Следующий уровень объединяет устройства в узлы микропроцессора: устройство управления, память, АЛУ и т. д. Такая многоуровневая система соединений при создании БИС реализуется в виде многослойной системы проводников. Выводы корпуса соединяются только с выходами и входами всей БИС и с точками подведения питания.

В связи с тем, что одним из факторов, ограничивающих сложность БИС, является ограниченное число выводов корпуса, возникает задача максимального сокращения числа выводов БИС посредством различных схемотехнических решений, оптимального разбиения систем на БИС, выбора компонентов и элементарных логических схем и технологии.

На рис. 1.8 приведены корпуса для полупроводниковых БИС. Они имеют нормализованный ряд конструкций и габаритов.

Корпуса для БИС выпускают с 16, 18, 24, 28, 40 и 48 выводами. Самый дешевый корпус для БИС пластмассовый.

В настоящее время уровень биполярной технологии можно охарактеризовать следующими достижениями: ширина базы транзистора 150 нм, граничная частота транзистора 6 ГГц, четыре уровня металлизации. Для маломощных ТТЛШ БИС достигнута интеграция 10 тыс. вентилей на кристалл, задержка 2,5 нс/вент., получены 8-, 16-разрядные процессорные секции. Применение беспороговой логики (аналогичной ЭСЛ) позволит получить еще более высокое быстродействие и плотность упаковки, а также малую рассеиваемую мощность (1 мВт/вент).

Модернизация изопланарной технологии позволила изготавливать БИС ЭСЛ с типовой задержкой вентилей 1,2 нс.

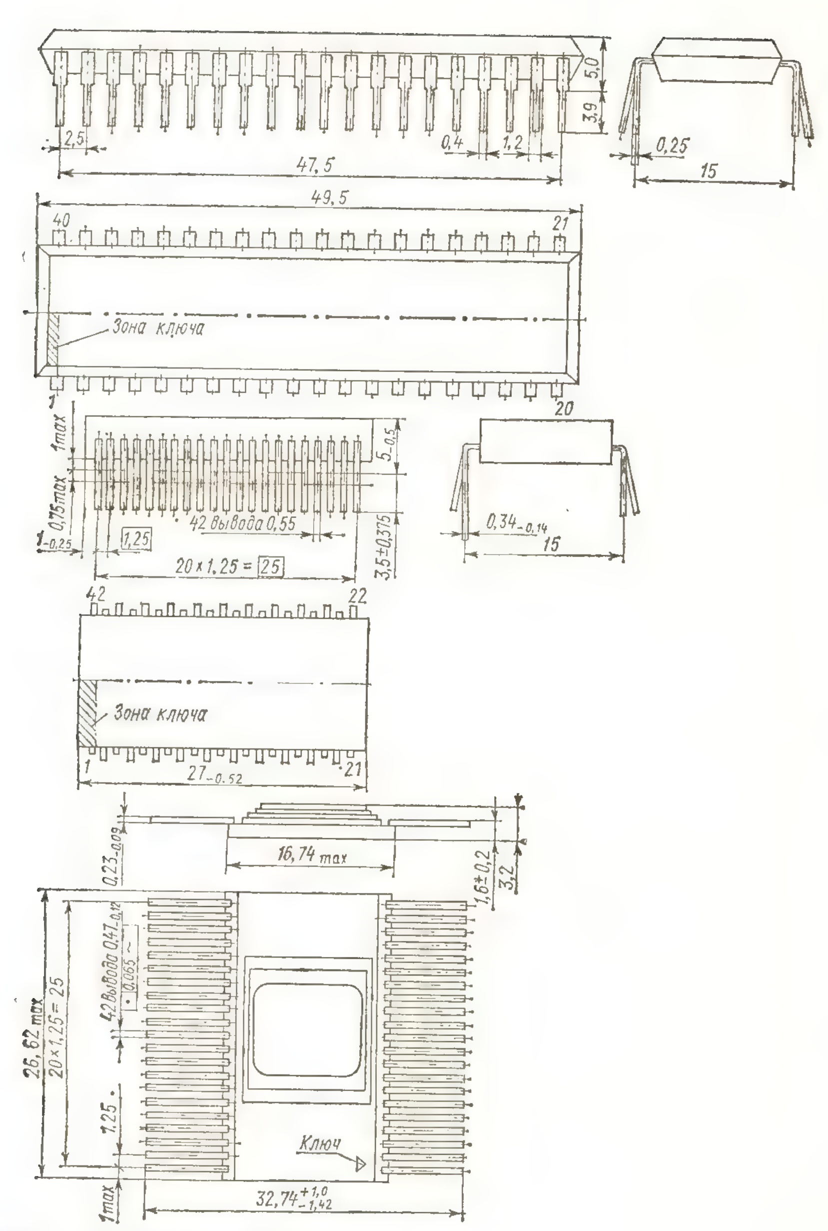


Рис. 1.8. Корпуса для микропроцессоров

Дальнейшее совершенствование биполярных БИС будет осуществляться с помощью технологических методов, разработанных применительно к МОП-приборам, применения структур кремний на диэлектрике и скрытых диэлектрических слоев. Вместо рекристаллизации осажденных кремниевых слоев выращиваются монокристаллические кремниевые пленки на монокристаллических подложках из шпинели (сложного соединения окислов магния и алюминия). Имплантируются атомы азота в кремниевые пластины на глубину примерно четверть микрона, в результате чего получается скрытый изолирующий слой. Применение способа комбинированного маскирования, при котором конфигурация всех критичных элементов интегральной схемы задается одновременно одной операцией маскирования, является основой новой технологии биполярных микропроцессоров с использованием ионного легирования. В этом случае уменьшаются допуски на совмещение разных масок, а следовательно, и компенсирующие запасы на любые возможные взаимные смещения этих масок друг относительно друга. В результате элементы схемы можно делать более компактными и располагать ближе друг к другу, уменьшаются ширина линий и размеры элементов, повышается плотность упаковки и снижаются паразитные емкости, что увеличивает скорость переключения схем. Технология позволяет получить БИС на кристаллах площадью до 50 мм^2 с задержками 1—2 нс/вент. при мощности потребления до 100 мВт на кристалл.

МОП-микропроцессоры. МОП-микропроцессоры имеют исключительно низкую рассеиваемую мощность (порядка милливатт в режиме переключения и нанаватт в статическом режиме). Сравнительно низкое быстродействие МОП-транзисторов обусловлено медленным перезарядом емкостной нагрузки, состоящей из собственной емкости транзистора и емкости схемы.

МОП БИС изготавливаются так же, как и биполярные по планарной технологии. При этом используют те же операции, что и при изготовлении биполярных микросхем, только технология изготовления МОП БИС по сравнению с биполярными проще.

Классификация МОП БИС (по технологическим признакам) приведена на рис. 1.9. Большинство модификаций обусловлено снижением порогового напряжения и повышением быстродействия МОП ИС. Более высокая подвижность электронов по сравнению с дырками приводит к тому, что скорость переключения n-канальных МОП БИС намного выше, чем p-канальных. К тому же низкий порог отпирания n-канальных приборов позволяет использовать источник питающего напряжения +5 В (стандартный для биполярных БИС).

Использование технологии ионного легирования для изготовления МОП БИС приводит к значительному уменьшению паразитных емкостей. Перекрытие затвором областей стока и истока снижается настолько, что емкости затвор—сток и затвор—исток становятся на порядок меньше, чем в диффузионных МОП БИС.

При использовании ионного легирования можно существенно уменьшить размеры областей истока и стока МОП-транзистора, в результате чего снижаются емкости исток—подложка и сток—подложка. Предельная частота МОП БИС, изготовленных методом ионного легирования, достигает 50 МГц.

В настоящее время применяют несколько методов изготовления МОП БИС с низким пороговым напряжением, а именно метод, основанный на использовании кремния, ориентированного в плоскости 100, а не в плоскости 111, как обычно; метод, при котором в качестве диэлектрика вместо двуокиси кремния используют нитрид кремния; метод, позволяющий применять кремниевый затвор вместо металлического и ряд других.

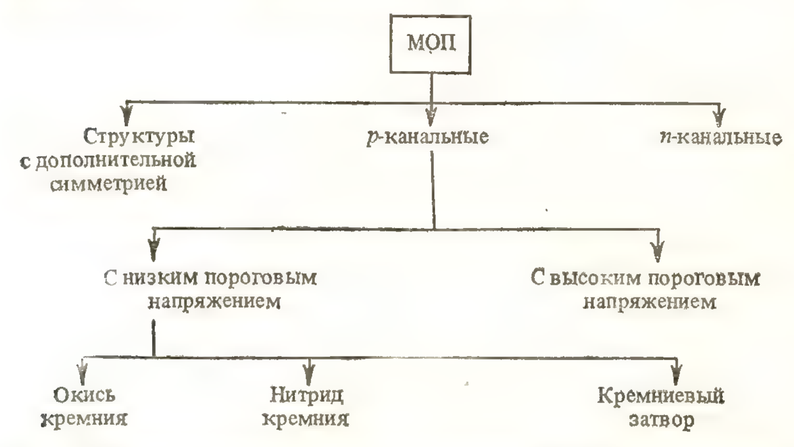


Рис. 1.9. Классификация разрабатываемых типов МОП БИС по технологическим признакам

В первом методе уменьшение порогового напряжения связано с тем, что поверхностный заряд в кремнии, ориентированном в плоскости 100, меньше по сравнению с зарядом в кремнии, ориентированном в плоскости 111. Поверхностный заряд обусловлен, в частности, наличием ненасыщенных связей между атомами на поверхности кристалла, а в плоскости 111 таких ненасыщенных связей больше, чем в плоскости 100.

При использовании нитрида кремния вместо двуокиси кремния пороговое напряжение уменьшается (а емкость затвора увеличивается), так как диэлектрическая проницаемость нитрида вдвое больше, чем проницаемость двуокиси кремния. Коэффициент усиления такой транзисторной структуры повышается на 50 %.

Третий метод заключается в использовании в качестве материала затвора сильнолегированного кремния вместо металла. Для создания кремниевых затворов применяют поликристаллический кремний, у которого работа выхода меньше, чем у алюминия, используемого в обычных МОП ИС. Это приводит к уменьшению

разности работ выхода материала затвора и подложки и, кроме того, к уменьшению поверхностного заряда. Все это снижает пороговое напряжение. Метод позволил снизить пороговое напряжение до 0,4 В, втрое увеличить быстродействие и уменьшить площадь схем почти в два раза. Структура транзистора с кремниевым затвором показана на рис. 1.10.

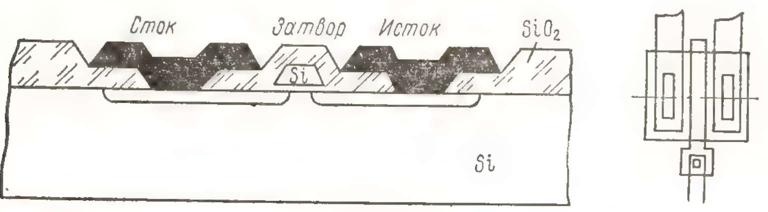


Рис. 1.10. Структура полевого транзистора с кремниевым затвором

Повышение быстродействия и снижение потребления мощности ИС достигается при создании структур с дополнительной симметрией (КМОП), т. е. структур, содержащих *p*- и *n*-канальные МОП-транзисторы на одной подложке, причем возбуждающий сигнал подается на затворы обоих приборов, соединенные вместе

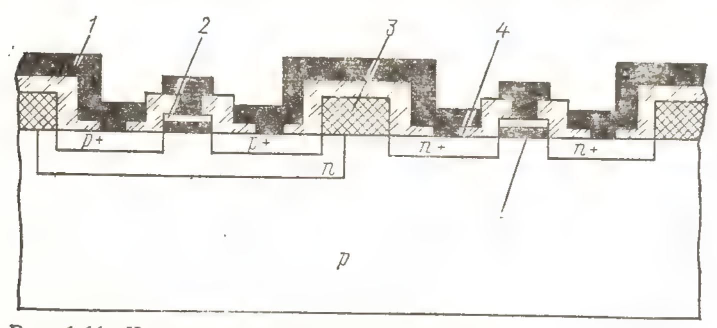


Рис. 1.11. Комплементарная МОП-структура:

1 — слой металлизации; *2* — затвор; *3* — толстый слой окиси; *4* — пассивирующий слой окиси; *5* — слой окиси затвора

(рис. 1.11). Логический элемент на КМОП-транзисторах в статическом состоянии практически не потребляет мощности. Если на входе нулевое напряжение, *p*-канальный транзистор открыт, а *n*-канальный закрыт, если на входе единица, то наоборот.

Общепринятая МОП-технология может послужить основой для создания КМОП-технологии, которую получают дополнением транзисторной структуры *n*-карманом, размещаемым в той же высокоомной подложке.

Для создания КМОП БИС чаще всего используется традиционная комплементарная логика. Она не требует генераторов, тактовых импульсов, имеет больший диапазон рабочих напряжений

и позволяет создать простые вентили с меньшим числом транзисторов.

Новая технология создания КМОП БИС с кремниевыми затворами позволила снизить задержку в затворе почти вдвое. Полученные приборы характеризовались задержкой в пределах от 2 до 3 нс при коэффициенте разветвления по выходу, равном 2.

КМОП БИС на сапфировых подложках представляют собой маломощные быстродействующие приборы. Их использование в радиоэлектронном оборудовании судов обусловлено высокой помехоустойчивостью, широким диапазоном рабочих температур и радиационной стойкостью.

Усовершенствованные методы МОП-технологии позволили получить однокристальные 16-, 32-разрядные микропроцессоры и микроЭВМ с плотностью элементов, достигающей 0,5 мил транзисторов в кристалле.

Для производства быстродействующих микропроцессоров началось применение арсенид-галлия.

Создание технологических арсенид-галлиевых кристаллов зависит прежде всего от чистоты материала подложки и стабильности свойств поверхности.

Разбросы пороговых напряжений от кристалла к кристаллу получаются от 5 до 10 %, что достигается благодаря правильно спроектированным структурам приборов, в которых активные слои находятся в объеме кристалла, под поверхностью с нестабильными характеристиками. Вентили на GaAs должны переключаться за 30 пс при потребляемой мощности всего около 150 мкВт.

Глава 2

МИКРОПРОЦЕССОРЫ В СУДОВОЙ РЭА

2.1. ОБЛАСТИ ПРИМЕНЕНИЯ

В настоящее время области применения МП значительно расширяются. Это обусловлено тем, что микропроцессорные БИС можно максимально приблизить к объекту управления. С момента появления МП начал осваивать новые «специальности», он стал основным звеном в обработке больших потоков информации, получаемых от различных источников. Эту информацию МП принимает, обрабатывает, хранит, либо выдает на внешние звенья управления. Современные суда постоянно пополняются вычислительными средствами различного класса, начиная от простых контроллеров и кончая мультипроцессорными системами и многомашинными комплексами [48]. МП повлиял на структуру судо-

вой РЭА, которой присущи такие черты вычислительных средств четвертого поколения, как модульность организации, микропрограммируемость, магистральность. Стало возможным создание судовых автоматизированных систем управления (АСУ) различного уровня. Нижними звеньями таких АСУ являются микропроцессорные контроллеры, осуществляющие, как правило, непосредственную связь с управляемыми объектами. Верхними — мощные вычислительные средства, в которых широко используются микропроцессорные БИС. Промежуточный уровень судовых АСУ занимают микроЭВМ и специализированные вычислители.

Благодаря МП автоматизированы такие трудоемкие процессы, как вычисление параметров движения судна, контроль его мореходно-прочностных характеристик, обработка гидро-радиолокационной информации, решение задач управления техническими средствами судна, диагностика и контроль при ремонтно-профилактических работах в условиях плавания и в доках [8, 60].

МП позволили создать судовые «интеллектуальные» терминальные комплексы. Развитые системы судовых АСУ совместно с «интеллектуальными» терминальными средствами служат ускорению перехода от вахтенного обслуживания к автоматической службе управления. Такая служба строится на базе АСУ различной степени сложности, начиная от простых систем сбора данных и «советчика» оператора, и заканчивая АСУ супервизорного цифрового управления. При создании судовых АСУ применяют разнообразные методы управления, такие как управление по возмущению (с упреждением) и адаптивное. Эффективная реализация этих методов невозможна без использования микропроцессорной техники.

Современный этап развития навигационной техники связан с созданием быстродействующих микропроцессорных систем. В последние годы на основе МП спроектированы несколько типов автоматизированных радиолокационных прокладчиков, приемоиндикаторов спутниковых систем и другие сложные навигационные комплексы. Такое оборудование предназначено, в первую очередь, для крупнотоннажных судов. Однако настало время внедрять вычислительную технику на средне- и малотоннажные суда, на которые нецелесообразно устанавливать сложные и дорогостоящие навигационные комплексы. Ведущую роль здесь сыграют микропроцессорные средства. На ряде судов используют средства, настроенные для решения задач автосчисления пути судна с расчетом текущих координат и расстояния до конечной точки, автосчисления пути судна по заданному расстоянию, вычисление координат судна по данным приемоиндикаторов (без спецкарт и таблиц), определения места судна по Солнцу и звездам. Производится также расчет расстояния при плавании по дуге большого круга (ДБК) или локсодромии и расчет координат промежуточных точек для прокладки ДБК на меркаторской карте.

Для выполнения автоматического счисления пути судна применяют внешние (по отношению к вычислительному средству) судовые устройства — судовые лаги, приборы времени, судовую сигнализацию [8].

По пройденному расстоянию, курсу судна, координатам исходной и конечной точек (точек поворота) определяют текущие координаты и расстояние до конечной точки. Возможность модификации программного (микропрограммного) обеспечения микропроцессорной навигационной аппаратуры позволяет варьировать способы управления и вид выводимой информации в широких пределах.

При периодических вычислениях счисляемые координаты судна выводятся на терминальные табло «Широта» и «Долгота». С прибытием судна на конечную точку или точку поворота включается звуковая и световая сигнализации.

Возможность программной перенастройки на решение других задач (либо аппаратные возможности МП) позволяет переходить из автоматических режимов (например, режима автоматического счисления) к решению другой задачи в пультовом режиме. Если данная задача является штатной в системе программного обеспечения, то необходим только ее признак (номер). В процессе эксплуатации состав штатных программ может меняться. При работе в режиме ЭВМ — оператор вводят величины (отсчеты приемоиндикаторов, секстана и др.), полученные из наблюдений. Микропроцессорное средство может представлять недостающую информацию, автоматически вычисляя ее по жестким алгоритмам.

При плавании по огражденным фарватерам микропроцессорное средство оповещает судоводителя специальной сигнализацией о прохождении траверзов знаков ограждения и о расстоянии до очередного буя или навигационного знака.

Определение места судна по Солнцу и звездам с помощью микропроцессорного вычислительного средства избавляет судоводителя от использования таблиц МАЕ, ВАС-58, МТ-75 и выполнения графических построений. При этом время обработки астрономических наблюдений сокращается более чем в 4 раза.

Микропроцессорное навигационное средство представляет собой, как правило, микроЭВМ, связанную с контроллерами навигационного оборудования.

Микропроцессорные средства успешно применяют в системах контроля мореходных качеств судна. Функциональное ядро этой системы составляет сеть каналов в активных и частично активизированных успокоительных цистернах, которые по существу являются единственным средством умерения качки судна в дрейфе или при малых скоростях хода. В эту систему входят: датчики, позволяющие определить период бортовой качки; сеть каналов, задерживающих стабилизирующую жидкость; управляющее микропроцессорное устройство (контроллер) и оконечные усилители.

При слабой и умеренной качке цистерны работают пассивно и колебания воды (стабилизирующей жидкости) в цистернах на 90° отстают от колебаний судна. В случае увеличения бортовой качки включается микропроцессорное устройство, активизирующее специальные воздушные каналы, удерживающие стабилизирующую жидкость в определенном положении. В результате колебания жидкости происходят по трапецеидальному закону (максимальное умерение качки достигает 70—90 %). Микропроцессорные устройства позволяют вести более плавное и гибкое управление клапанами каналов. Кроме того, микропроцессорное устройство может осуществлять контрольно-диагностические работы.

Интенсивное пополнение флота контейнеровозами, ролкерами, лихтеровозами и другими специализированными судами существенно изменило характер грузовых операций. Масса одного крупногабаритного грузового места исчисляется многими десятками, а иногда и сотнями тонн. Однако стоянки этих судов в портах под погрузкой или выгрузкой занимают всего несколько часов. Во время таких грузовых операций проводится тщательная проверка мореходных качеств судна, таких как остойчивость, крен, дифферент, прочность корпуса. В связи с этим необходимо постоянно анализировать правильность размещения грузов, а зачастую осуществлять балластировочные операции. Основным документом для получения всех характеристик остойчивости служит «Информация об остойчивости для капитана». Существующие методы контроля остойчивости являются трудоемкими и не гибкими.

В настоящее время создается микропроцессорная аппаратура контроля остойчивости судна, предназначенная для дистанционного измерения осадки судна и судовых запасов. Такая аппаратура входит в отдельную автоматическую систему контроля остойчивости и прочности судна. Аппаратура контроля включает в себя датчики осадки, устанавливаемые в кормовых и носовых отсеках судна, датчики уровня, размещаемые во всех судовых танках, и микропроцессорное обрабатывающее устройство. В любой момент стоянки судна в порту и обязательно перед выходом в рейс дается оценка начальной остойчивости. Остойчивость судна и судовые запасы во время движения проверяют непрерывно. Для этого производят расчет метацентрической высоты судна в зависимости от приращения угла крена, объема воды, необходимой для накрена на угол α, объемного водоизмещения и расстояния от центра тяжести до диаметральной плоскости. Входными величинами для расчета являются показатели уровнемеров танков судна до и после накрена, показатели датчиков осадки, данные грузового размера судна. В зависимости от типа судна указанные величины будут различными. Так что создание универсальной аппаратуры контроля без микропроцессоров потребует значительного объема оборудования, снижающего общую надежность работы системы.

Автоматизация контроля остойчивости имеет особо важное значение при аварийных ситуациях, когда судно утратило некоторый запас плавучести. Время и простота пользования здесь имеют жизненно важное значение. Скоротечность изменения обстановки при авариях требует, чтобы метод контроля был точным, обеспечивал быстрое и непрерывное получение данных, на основании которых определяют критический момент состояния остойчивости и принимают меры, необходимые для ведения борьбы за живучесть судна.

Примером применения микропроцессорных БИС в системах контроля мореходных качеств судна являются контроллеры уровнемера. Например, уровнемер РУМБ-БК обеспечивает измерение уровня жидких продуктов (нефти, сжиженного газа) в резервуаре с погруженным стержнем. Уровень продукта фиксируется поплавком с постоянным магнитом. Пьезоэлектрический возбудитель формирует прямой (стартовый) сигнал, в результате чего возникает волна, распространяющаяся по стержню. Под воздействием магнита изменяются магнитострикционные свойства стержня, что вызывает появление отраженного сигнала при прохождении границы раздела двух сред. Второй, отраженный сигнал формируется у дна резервуара. Анализируя временные интервалы между отраженным и стартовым сигналами, габаритные размеры стержня, скорость распространения сигналов, температуру окружающей среды, можно с высокой точностью (до 20 мм) определить уровень жидкости. Малые размеры, возможность реализации сложных алгоритмов, простота сопряжения с различным оборудованием позволяют МП стать основой элементной базы в системах контроля мореходных качеств судна.

Существенна роль микропроцессорной аппаратуры и при обработке гидро-радиолокационной информации. Это связано, в первую очередь, с огромным объемом поступающей информации и с переложением целого ряда функций человека-оператора на вычислительные микропроцессорные средства. С применением МП реализованы анализаторы помех, приборы управления каналов измерения, приборы первичной и последующих обработок информации, блоки цифровой фильтрации, блоки управления визирами, блоки корреляции и стабилизации потоков информации, взвешивания входных последовательностей, устройства вычисления центральной доплеровской функции и др.

В настоящее время появляются новые функциональные задачи, изменяются методы организации управления и решения задач. Применение микропроцессорных вычислительных средств позволило эффективно производить анализ спектральной характеристики с использованием быстрого преобразования Фурье, когерентной фильтрации, корреляционной обработки.

Современные методы исследования морского дна, например для разведки полезных ископаемых, не мыслимы без широкого использования микропроцессорных систем. Причем для решения слож-

ных задач гидролокации приходится создавать многопроцессорные комплексы. Характерным примером может служить судно гидроразведки. При гидролокации морского дна с кормовой части судна выбрасывается «лапа», снабженная большим (в зависимости от параметров исследуемого района) количеством датчиков акустических сигналов. На судне устанавливается мощный источник акустических импульсных колебаний. Последние, распространяясь в воде, доходят до дна, отражаются и поступают на датчики «лапы». После обработки параметров отраженных сигналов математическими методами можно получить реальную картину морского дна, определить наличие нефтяных и газовых залеганий. Для обеспечения эффективного «просмотра» необходима быстродействующая вычислительная система, обеспечивающая первичную обработку акустической информации с каждого датчика и выдачу результатов в вычислитель верхнего звена обработки, а также на систему отображения информации.

Эта система преобразовывает результаты в наглядную форму (выдача рисунка на графопостроитель), а также определяет аномальные придонные объекты. Такая система должна обладать высоким быстродействием и широкими вычислительными возможностями. Средства первичной обработки информации должны быть максимально приближены к датчикам и программно доступны вычислителю верхнего звена.

Микропроцессорные средства находят применение и в контрольнодиагностической аппаратуре [60]. Повышение уровня автоматизации судов, насыщенности их сложными вспомогательными механизмами обусловливает новые задачи технического обслуживания, которые связаны, с одной стороны, с усложнением судового оборудования, с другой, — с сокращением численности экипажа. В то же время возможности баз технического обслуживания ограничены. Так, например, по имеющимся данным, даже постоянное пребывание на каждом лихтеровозе бригады максимального состава позволяет выполнить в среднем только 23,8 % работ по техническому обслуживанию. На судах типа «Юлиус Фучик» экипаж в состоянии выполнить только 39,9 % работ по техническому обслуживанию. Вместе с тем экономически не выгодно для обеспечения подобных работ выводить суда из эксплуатации. Принимаемые меры по увеличению продолжительности рабочего дня не являются кардинальным решением задачи обслуживания. Необходимы простые, надежные и эффективные методы безразборного контроля. Из-за большой разновидности судовых технических средств требуется дифференцированный подход к применяемым методам технической диагностики. Так целый ряд средств должен подвергаться постоянному контролю. Это касается, в первую очередь, главных и вспомогательных энергетических установок. Большое количество вспомогательных механизмов необходимо контролировать периодически. В обоих случаях контролирующие функции можно переложить на микро-

процессорные устройства управления, снабженные соответствующими датчиками. Такие устройства можно легко унифицировать, придавая им различные постоянные запоминающие устройства программ для соответствующего типа оборудования.

Постоянная работа систем диагностики и оптимизации режимов работы главного двигателя только на танкере «Поль Робсон» экономит 15 т топлива ежегодно. Увеличивается и срок службы двигателя [61].

Разработаны и начинают выпускать микропроцессорные средства безразборного контроля. Безразборные методы контроля и диагностики существенно сокращают число сеансов разборки механизмов, которые сами по себе являются нежелательными. Нарушение сопряжений приработанных поверхностей приводит ко вторичной их приработке, износ при которой соответствует износу при нормальной эксплуатации в течение сотен часов. Опыт разборки вспомогательных механизмов экипажем показывает, что при этих операциях имеют место случаи повреждения деталей и узлов механизмов, а также выходят из строя элементы крепежа, прокладки, сальниковые уплотнения и др. При этом возможно попадание грязи в механизмы. Следовательно, нужно максимально избегать разборки механизмов. Например, на судах Черноморского пароходства осуществляется контроль микропроцессорными средствами технического состояния основных элементов гидравлических рулевых машин без разборки и дефектации [60].

Перевод вспомогательного оборудования на широкое использование микропроцессорных средств технической диагностики позволит сократить трудозатраты на обслуживание и ремонт, увеличить срок службы механизмов и качественно повысить уровень обслуживания и глубину контроля.

Трудно перечислить все оборудование, где могут использоваться БИС. Выше были описаны характерные случаи их применения.

В дальнейшем приведены конкретные примеры применения микропроцессорных БИС в контроллерах, серийных и специализированных микроЭВМ и микропроцессорных системах.

2.2. КОНТРОЛЛЕРЫ

Применение микропроцессорных комплектов эффективно при построении контроллеров. *Контроллер* — это автомат, работающий по определенному закону и выполняющий функции управления каким-либо периферийным или автономным устройством в соответствии с заданным алгоритмом при поступлении на его вход некоторого набора команд.

Контроллеры первого поколения решали задачи, ориентированные в основном на управление вполне определенным периферийным устройством. Появление МП и другой современной эле-

ментной базы дало возможность перейти к конструированию контроллеров, отличающихся значительно большими управляющими возможностями и универсальностью.

Контроллеры дополнились элементами ЭВМ. В них появились блоки постоянной и оперативной памяти, что позволило закладывать соответствующую программу управления. При необходимости ее можно изменять.

Включение в контроллеры малогабаритных микропроцессорных блоков обработки данных и интерфейсных блоков позволяет

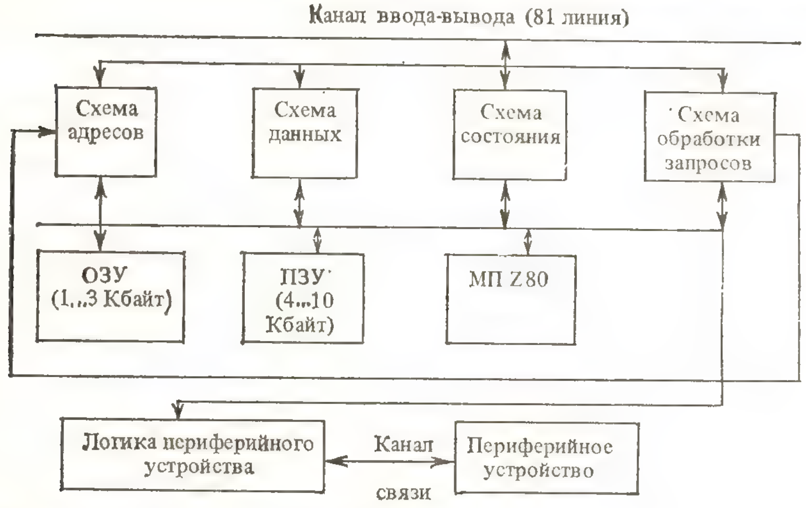


Рис. 2.1. Структурная схема контроллера «Серия 1»

им обеспечивать управление и связь с разнотипными периферийными устройствами, такими как дисплеи, накопители информации на магнитных лентах и дисках, цифропечатающие устройства и др.

Контроллеры на базе микропроцессорных БИС делят на специализированные, программируемые и технологические. Специализированные контроллеры могут управлять несколькими более простыми контроллерами и по структуре и функциональным возможностям аналогичны специализированным микроЭВМ нижних уровней управления. Они имеют центральное устройство обработки информации, запоминающие устройства малого объема и схемы обработки прерываний.

Типичным специализированным контроллером является прибор Серия 1, применяемый в судовых технических средствах Финляндии и Швеции (рис. 2.1).

Контроллер осуществляет прием и первичную обработку информации с *периферийных устройств* (датчиков, судовых приборов и механизмов). Результат может быть направлен как в централь-

ную ЭВМ, так и на *периферийные устройства*. Контроллер построен на базе МП Z80 фирмы «Зайлог» (США) [48].

Обмен с центральной ЭВМ осуществляется посредством четырех специализированных схем, включающих средства диагностики.

Во все регистры схем можно записать, а затем считать тестовую информацию. *Схема адресов* управляет потоком данных при обмене типа запоминающее устройство (ЗУ) — периферия (управление осуществляется в соответствии с инструкциями, поступившими от МП). Эта схема может некоторые простые операции выполнять без вмешательства МП, но под контролем центрального процессора (ЦП) управляющей машины. *Схема данных* служит для кратковременного запоминания данных и обмена либо с ЗУ ЦП, либо с шиной МП. При передаче данных производится контроль по четности. *Схема состояния* используется для передачи информации о работе периферийного устройства, кодах прерываний, состояний ЦП. *Схема обработки запросов* выставляет запросы на обслуживание периферийных устройств в *канал ввода-вывода* и следит за готовностью ЦП к реализации запросов.

При работе внешнего устройства в режиме прямого программного управления ЦП непосредственно управляет работой этого устройства, прерываясь для выполнения операций ввода-вывода. При работе в режиме «Занятия» осуществляется прямой доступ в ОЗУ управляющей ЭВМ. Во втором режиме МП только занимает циклы памяти, не прерывая работу ЦП.

Наряду со специализированными контроллерами, достаточно сложными по своей структуре и решаемым задачам, проектируют более простые, выполняющие ограниченные задачи и функции.

К ним относятся так называемые программируемые и технологические контроллеры.

Программируемые контроллеры могут быть использованы для некоторых классов двигательных механизмов, контрольно-измерительных приборов, средств связи. Каждый из таких контроллеров может заменить от 1 до 5 аналого-механических регуляторов.

В качестве примера можно привести разработанный фирмой «Сигнетикс» (США) одноконтурный контроллер, предназначенный для обмена информацией между управляемым оборудованием (при однобитовых данных) и центральной управляющей микроЭВМ

Этот контроллер построен на базе микропроцессоров 8080 и схемы 8251. Данные, поступающие побайтно на вход схемы, преобразуются в однобитовые посылки (преобразование типа параллельный — последовательный код и наоборот) в соответствии с режимом передачи, код которого поступает с МП. Инициализация обмена может быть осуществлена по линии «Прерывание».

Контроллер позволяет вести обмен по асинхронным и синхронным последовательным или параллельным каналам, что значительно расширяет область его применения.

Технологические контроллеры являются многоконтурными, управляют несколькими группами объектов управления и могут заменить от 5 до 50 аналоговых регуляторов.

На рис. 2.2 изображена структурная схема технологического контроллера, входящего в систему ТДС 2000, разработанного фирмой «Хонеэл» (США).

Контроллер этой системы включает в себя МП, 8 БИС *ОЗУ* с организацией 256×4 бит каждая, 23 БИС *ППЗУ* по 512×8 бит каждая, БИС 16-разрядного *мультиплексора*, а также сдвоенные преобразователи с широтно-импульсной модуляцией и регистрами запоминания управляющих сигналов.

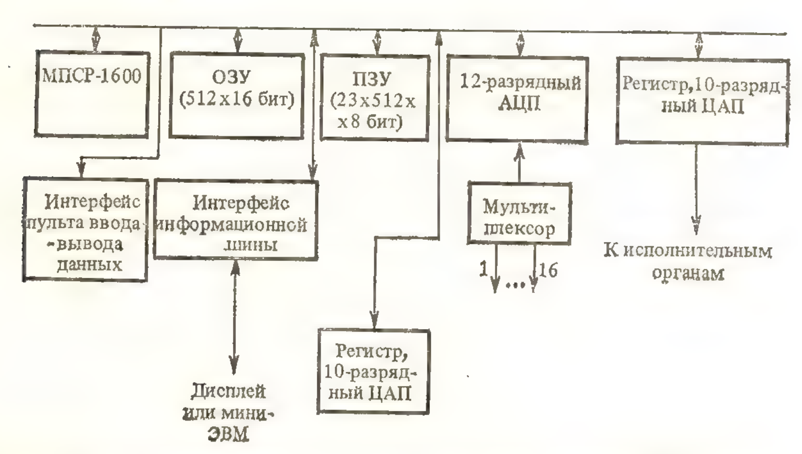


Рис. 2.2. Структурная схема контроллера, входящего в систему ТДС 2000

Контроллер может менять нижние и верхние уставки изменения параметров и организовывать связь с центральным процессором ЭВМ-ядра. Программа работы контроллера, записанная в ПЗУ, может быть изменена. В поставляемом варианте в эту программу входят 28 алгоритмов управления.

Из отечественных разработок можно привести контроллер АОШ (адаптер «Общая шина»), предназначенный для сопряжения внутренней магистрали (ВМ) микроЭВМ с магистралью «Общая шина» (ОШ). Конструктивное исполнение контроллера и технология изготовления обеспечивают его эксплуатацию в судовых условиях. Контроллер АОШ обеспечивает выполнение следующих функций:

осуществляет передачу адресов и данных между магистралями при выполнении операций записи, чтения, чтения с модификацией, при этом задатчик может находиться как на магистрали ВМ, так и на магистрали ОШ;

формирует интерфейсные сигналы при управлении обменом информацией между магистралями;

транслирует запросы прерывания по четырем уровням из магистрали ОШ в магистраль ВМ;

транслирует разрешения прерывания по четырем уровням;

осуществляет передачу вектора прерывания из магистрали ОШ в магистраль ВМ.

Приведенные примеры показывают, насколько широк диапазон задач, решаемых контроллерами. Эта аппаратура отличается высокой экономичностью, достигаемой благодаря применению микропроцессоров.

Внедрение контроллеров может идти как по пути простой замены устаревшего оборудования (при уменьшении массогабаритных характеристик в целом), так и по пути принципиальных изменений системного плана. В частности, внедрение контроллеров в значительной мере может повлиять на структуру управления в сторону ее децентрализации.

Внедрение контроллеров на базе МП позволит сделать качественный скачок в развитии различных судовых систем.

2.3. МИКРОЭВМ

Судовые микроЭВМ подразделяют на специализированные и широкого применения [14]. Специализированные имеют, как правило, нетрадиционную структуру и специфическую систему команд. Они предназначены для решения узкого круга задач. К ним относятся, например, задачи оптимизации работы судовых энергоустановок в зависимости от количества и характеристик нагрузок, вида топлива, электрических характеристик тракта передачи энергии и др. [42].

Такие микроЭВМ имеют разрядность 8 или 16 бит, размещаются на одной, либо нескольких платах. Система команд обычно состоит из 20 ... 50 команд. Система команд формируется как подмножество команд ЭВМ широкого применения, либо из узкоспециализированных команд, характерных для данного класса задач.

МикроЭВМ широкого применения имеют типичную структуру фон-Неймана и одну из распространенных систем команд. Они представляют собой многоплатные конструкции, предназначенные в основном для решения задач вычислений и обработки данных, одноплатные (однокристальные) встраиваемые средства, ориентированные на работу в составе управляемого оборудования и приборов, и персональные микроЭВМ (ПЭВМ).

Многоплатными были первые отечественные и зарубежные микроЭВМ «Электроника С-5», «Электроника НЦ», «Электроника-60», «LSI-11», «Nova», «Rolm» и др. Они имели различные системы команд, были несовместимы программно и аппаратно. Многоплатные микроЭВМ, сопряженные со стандартными периферийными устройствами, образуют единый вычислительный комплекс, размещаемый в стойке или специальном столе (СМ-1800, 15ВМ-16, 15ВУМС-28-025).

Одноплатные микроЭВМ — результат дальнейшей миниатюризации вычислительных средств.

Их аппаратура размещается на одной плате («Электроника МС»), а вычислительные возможности сравнимы с многоплатными. К этой категории можно отнести однокристальные микроЭВМ. Они размещаются на одном кристалле и имеют все основные элементы ЭВМ. Однако выпускаемые однокристальные микроЭВМ имеют уникальные системы команд, не совместимые с системами широкого применения (К1814, К1816, К1820, К1803). Следует отметить, что для обеспечения процесса разработки программ в комплект однокристальных микроЭВМ входят эмуляторы (К1803ВЕ2), кристаллы с электрически программируемыми запоминающими устройствами, либо предусмотрены средства для подключения внешней памяти.

МикроЭВМ управляют группами контроллеров, потоками информации, выполняют сложные вычисления. Они могут работать практически на любом уровне систем управления.

Персональные микроЭВМ («Электроника Т3-29 МК») появились в СССР в 1983 г. и сразу обратили внимание разработчиков судовой РЭА. Их отличают:

развитый человеко-машинный интерфейс, обеспечивающий простое и наглядное управление ПЭВМ непрофессиональным пользователем;

большое число готовых программных средств прикладного характера;

малогабаритные накопители информации значительной емкости на сменных носителях, обеспечивающие взаимозаменяемость и эксплуатацию приобретаемых программных средств;

малые габариты и масса, позволяющие устанавливать ПЭВМ на любом рабочем месте.

Персональный характер ПЭВМ понимается не столько в плане личной принадлежности, сколько в плане возможности эксплуатировать ее без помощи профессионального программиста.

Отечественной промышленностью выпускается несколько типов ПЭВМ — «ЕС 1840», «Нейтрон И9.66», «Электроника Т3-29 МК», «Искра 1030.11», «Океан 240.2».

2.4. МИКРОПРОЦЕССОРНЫЕ СИСТЕМЫ

Судовые микропроцессорные системы предназначены для решения широкого класса задач, связанных с управлением движения, контроля живучести и безопасности судна, управлением работой современных радиолокационных и гидроакустических станций.

Мультипроцессорные системы, построенные на МП, одновременно позволяют обеспечивать возможность работы в реальном масштабе времени, высокую надежность и готовность, хорошую гибкость и модернизационную способность. Однако самым главным достоинством мультипроцессорных систем является их вы-

сокая пропускная способность, достигаемая путем распределения нагрузки между отдельными микропроцессорами и благодаря этому параллельная обработка информации. В зависимости от характера решаемых задач вычислительную систему можно строить на однотипных или разных микропроцессорах.

При необходимости в такие системы можно включать специализированные микропроцессоры, ориентированные на выполнение вполне определенного класса задач, таких как быстрое преобразование Фурье, преобразование координат, вычисление тригонометрических функций, выполнение разного рода групповых операций и т. п.

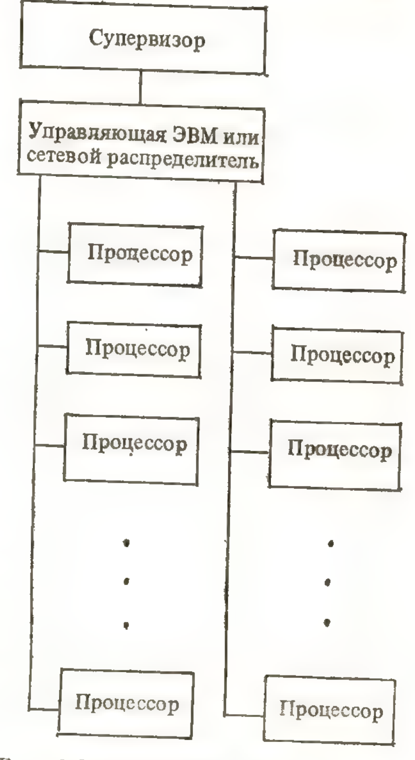


Рис. 2.3. Матричная вычислительная система

Создание малогабаритных процессоров и микроЭВМ дает возможность широко применять в судовой технике многопроцессорные системы с реконфигурируемой структурой. Отличительными особенностями реконфигурируемых структур являются их высокие надежность и готовность. Реконфигурируемые мультипроцессорные системы имеют несколько процессоров, как правило, одного типа. На каждый из процессоров возлагается решение вполне определенных задач или эпизодов. Обычно в такой системе имеется один или несколько резервных вычислительных модулей.

В зарубежной практике в настоящее время широкое распространение получили так называемые квазимультипроцессорные системы, в частности такие, как система с параллельным соединением процессоров, система поточной обработки информации и т. п.

Одной из перспективных систем для параллельной обработки больших полей данных являются системы с матричными процессорами (матричная ЭВМ). Структура такой системы изображена на рис. 2.3.

Микропроцессоры нашли широкое применение в зарубежных гидроакустических системах в связи с возникшей необходимостью оперативного изменения рабочей частоты, мощности, длительности и формы зондирующих сигналов, размеров

и формы гидроакустических антенн, методов обзора пространства.

В системе формирования диаграммы направленности данные поступают в накопитель, который обновляется применительно к каждой выборке. В интервале между выборками происходит считывание с учетом расположения соответствующего элемента в антенне. Формирование лучей осуществляется последовательно, согласно принятому порядку. Имеющиеся в настоящее время статические ОЗУ и умножители на БИС функционируют при длительности такта менее 70 нс, что позволяет формировать до 100 лучей в полосе частот 70 Гц.

Используется только один умножитель для ввода весовых коэффициентов при формировании нескольких лучей, что уменьшает количество адресов и управляемых логических элементов. В качестве такого умножителя применяют умножитель-сумматор, выполненный на микропроцессорных БИС.

Большая гибкость систем формирования диаграммы направленности достигается применением управляющей памяти микрокоманд.

Система формирования 160 лучей 16-элементной антенны состоит из нескольких параллельно работающих специальных процессоров, управляемых микроЭВМ.

Широкие возможности дает применение МП в системе дистанционного автоматизированного управления главным дизелем судна. Система обеспечивает пуск, реверс и задание требуемой частоты вращения двигателя с мостика [47].

Основные достоинства системы контроля и управления на микропроцессорах следующие:

сокращение объема электронной аппаратуры на 65 %;

уменьшение количества элементов аппаратуры на 50 %, что вместе с сокращением длины соединительных проводов позволяет ожидать значительного повышения надежности и сокращения стоимости системы;

высокая точность результатов операций с аналоговыми сигналами благодаря использованию цифровых методов их обработки, что, например, полностью устраняет температурный дрейф и ряд других источников ошибок, присущих аналоговым способам обработки данных.

Глава 3

МИКРОПРОЦЕССОРНЫЙ КОМПЛЕКТ БИС СЕРИИ КР580

3.1. ОБЩИЕ СВЕДЕНИЯ

МПК БИС серии КР580 построен на базе *n*-МОП технологии и предназначен для цифровой РЭА средней производительности. Наиболее широко МПК БИС серии КР580 применяют в устрой-

Таблица 3.1. **Состав комплекта БИС серии КР580**

Обозначение ИС	Наименование	Напряжение питания, В	Тип корпуса
КР580ИК80А	Центральное процессорное устройство (микропроцессор)	+5; —5; +12	2123.40-2
КР580ИК51	Программируемый последовательный интерфейс	+5	2123.40-2
КР580ВИ53	Программируемый таймер	+5	2123.24-3
КР580ИК55	Программируемый параллельный интерфейс	+5	2123.40-2
КР580ИК57	Программируемый контроллер прямого доступа к памяти	+5	2123.40-2
КР580ВН59	Программируемый контроллер прерывания	+5	2121.28-5

ствах контроллерного класса. Состав комплекта приведен в табл. 3.1 [1, 2, 29, 34, 45]. Отличительными особенностями комплекта являются: программируемость на уровне команд, широкий набор специализированных БИС, развитое программное обеспечение, невозможность каскадирования центральных процессорных устройств, 8-разрядная организация шины данных, временно́е мультиплексирование информации.

До 1984 г. в обозначении выводов БИС комплекта была распространена отечественная аббревиатура. С введением в действие стандарта, регламентирующего порядок построения обозначений выводов и сигналов, используют буквы латинского алфавита.

При описании БИС комплекта приводится соответствие обозначений выводов и сигналов.

Периферийные БИС комплекта программируемы. Командные инструкции заносятся микропроцессорной БИС по 8-разрядной шине данных. Указанные БИС могут работать автономно при соблюдении необходимых условий. Далее рассмотрены шесть БИС, входящих в МПК.

3.2. ЦЕНТРАЛЬНОЕ ПРОЦЕССОРНОЕ УСТРОЙСТВО КР580ИК80А

Микросхема КР580ИК80А представляет собой однокристальное 8-разрядное центральное процессорное устройство (микропроцессор) с фиксированной системой команд. Содержит 16-разрядную трехстабильную шину адреса, 8-разрядную двунаправленную трехстабильную шину данных, 4 входных и 6 выходных управляющих выводов, имеет двухфазовую синхронизацию, внутреннее регистровое ЗУ, фиксированный набор выполняемых команд. Каждая команда занимает от одного до пяти машинных циклов (МЦ), каждый из которых состоит из 3 ... 5 периодов синхроимпульсов, так называемых машинных тактов (МТ).

Условное графическое обозначение микросхемы приведено на рис. 3.1. Структурная схема БИС дана на рис. 3.2, назначение выводов — в табл. 3.2.

Ниже даны латинские и соответствующие отечественные обозначения выводов и сигналов микросхемы КР580ИК80А: *INTE*—РПр, *WAIT* — Жд, *DBIN* — П, *READY*—Г, *F*1—Ф1, *F*2—Ф2, *RESET*—*R*, *INT*—ЗПр, *HOLD* — Зх, *WR*—В, *SYNC*—*F*, *HLDA*—ПЗх.

Рис. 3.1. Условное графическое обозначение микросхемы КР580ИК80А

Шина адреса *ША* обеспечивает адресацию внешней памяти объемом до 64 Кбайт, а также возможность обращения в одному из 256 устройств ввода или вывода. *А0* — младший разряд адреса.

Шина данных *ШД* обеспечивает двунаправленный обмен информацией между МП и внешними устройствами в режиме временного мультиплексирования. В первом такте каждого МЦ через эту шину выдается информация о состоянии (8 разрядов) МП. Она отображает действия, которые будут производиться схемой в текущем МЦ. *Д0* — младший разряд данных. Выходной сигнал *Синхро* определяет начало МЦ. Во время его действия МП по шине данных выдает информацию о состоянии схемы. Сигнал может быть использован для занесения этой информации во внешний регистр.

Выходной сигнал *Прием* указывает внешним устройствам, что шина данных находится в режиме приема. Выходной сигнал *Выдача* стробирует информацию, выдаваемую МП на шину данных, в результате чего информация остается стабильной во время действия сигнала.

Входной сигнал *Готовность* сообщает о готовности внешнего устройства вести обмен с МП. Если после выдачи адреса на адресную шину МП получил на вход *Г* напряжение низкого уровня, то он переходит в режим «Ожидание», в котором микросхема будет находиться до появления по линии *Готовность* напряжения высокого уровня.

Входной сигнал *Захват*, поступающий асинхронно на соответствующий вывод, переводит шины адреса и данных МП в состоянии «Останов» в высокоимпедансное состояние, что позволяет внешним устройствам занимать эти шины для инициализации

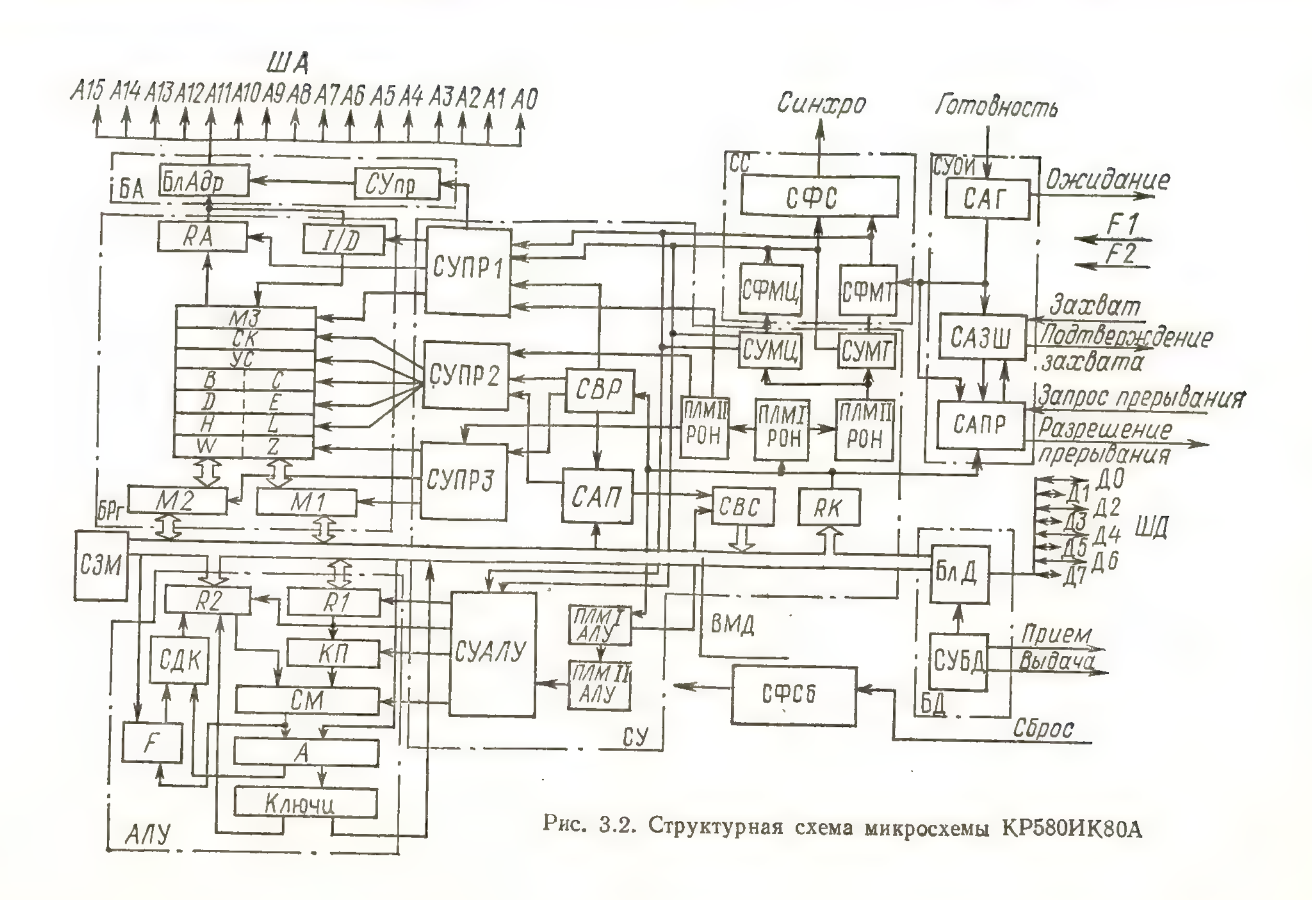


Рис. 3.2. Структурная схема микросхемы КР580ИК80А

Таблица 3.2. **Назначение выводов микросхемы КР580ИК80А**

Номер вывода	Назначение	Разряд	Обозначение	Тип
1	Шина адреса	10	A10	Выход
2	Общий	—	*GND*	—
3	Шина данных	4	Д4	Вход-Выход
4		5	Д5	
5		6	Д6	
6		7	Д7	
7		3	Д3	
8		2	Д2	
9		1	Д1	
10		0	Д0	
11	Питание 3 (—5 В)	—	$U_{CC}3$	—
12	Сброс	—	*R*	Вход
13	Захват	—	Зх	
14	Запрос прерывания	—	ЗПр	
15	Фаза 2	—	Ф2	
16	Разрешение прерывания	—	РПр	Выход
17	Прием	—	П	
18	Выдача	—	В	
19	Синхро	—	Сн	
20	Питание 2 (+5 В)	—	$U_{CC}2$	
21	Подтверждение захвата	—	ПЗХ	Выход
22	Фаза 1	—	Ф1	Вход
23	Готовность	—	Г	Вход
24	Ожидание	—	Жд	Выход
25	Шина адреса	0	A0	Выход
26		1	A1	
27		2	A2	
28	Питание 1 (+12 В)	—	$U_{CC}1$	—
29	Шина адреса	3	A3	Выход
30		4	A4	
31		5	A5	
32		6	A6	
33		7	A7	
34		8	A8	
35		9	A9	
36		15	A15	
37		12	A12	
38		13	A13	
39		14	A14	
40		11	A11	

Примечание. Шины данных и адреса трехстабильные; сигналы фаз Ф1 и Ф2 должны иметь амплитуду 12 В.

обмена. Сигнал *Захват* воспринимается МП в состоянии «Останов», в машинном такте *Т2* или в такте «Ожидание» ($T_{ож}$) при наличии на входе *Г* сигнала «1».

Выходной сигнал *Подтверждение захвата* является ответом МП на сигнал *Захват*. Он появляется при наличии напряжения высокого уровня на входе *Зх* по положительному фронту импульса синхронизации *Ф1* в тактах Т3 для МЦ «Чтение памяти» и «Чтение внешнего устройства» и следующим за Т3 (для МЦ в «Запись в память» и «Запись во внешнее устройство»).

Сброс сигнала *Подтверждение захвата* осуществляется по положительному фронту импульса синхронизации *Ф1* при наличии на входе *Зх* «0».

Входной сигнал *Запрос прерывания* используется внешними устройствами для запроса на обслуживание. Сигнал может поступать асинхронно. Он анализируется только в конце текущей команды или в состоянии «Останов».

Выходной сигнал *Разрешение прерывания* подтверждает внешний запрос прерывания. Этот сигнал отражает состояние внутреннего триггера *Разрешение прерывания*. Триггер устанавливают только программно. Сброс триггера осуществляется программно положительным фронтом импульса *Ф2* в такте Т1 МЦ «Подтверждение прерывания» либо подачей на вход *R* сигнала «1».

Входной сигнал *Сброс* прерывает выполнение команды МП, сбрасывает счетчик команд и регистр команд в нулевое состояние, переводит внутренние триггеры «Разрешения прерывания» и «Подтверждения захвата» в нулевое состояние, выводит МП из состояния «Останов» или «Захват», при этом содержимое аккумулятора, *РОН* и регистра условий не изменяется.

Во время действия сигнала *Сброс* шины адреса и данных находятся в третьем состоянии, а все выходные сигналы управления переходят в пассивное состояние. Сигнал *Сброс* должен длиться не менее трех периодов тактовой частоты.

На входы *Ф1* и *Ф2* поступают две серии неперекрывающихся импульсов синхронизации.

В состав структурной схемы входят следующие функциональные узлы (см. рис. 3.2): арифметико-логическое устройство *АЛУ*, схема синхронизации *СС*, схема управления *СУ*, схема управления обменом информации *СУОИ*, буфер адреса *БА*, буфер данных *БД*, схема формирования сброса *СФСб*, внутренняя магистраль данных ВМД, блок регистров *БРг*.

Параллельное 8-разрядное *АЛУ* выполняет арифметические и логические операции, операции сдвига над двоичными данными, представленными в дополнительном коде, а также может использоваться для обработки двоично-десятичных чисел. В состав *АЛУ* входят:

регистры *R1* и *R2* (8-разрядные). Предназначены для приема операндов с *ВМД*. *R2* используется также для выдачи результата операции на *ВМД*;

кодопреобразователь *КП*. Служит для передачи содержимого *R1* на сумматор в прямом и обратном коде;

комбинационный сумматор *СМ*. Выполняет операции сложения, вычитания и логические, оперирует 8-разрядными данными;

аккумулятор *А*. Предназначен для приема и хранения 8-разрядных данных в *СМ* либо с *ВДМ*;

ключи. Позволяют передавать содержимое *А* на *R2* или на *ВМД*;

регистр условий *F*. Служит для выдачи признаков результата операций из *СМ* на *ВДМ* либо на схему десятичной коррекции. Включает пять триггеров: знака (*S*), переноса (*С*), вспомогательного переноса (*С'*), четности (*Р*), нуля (*Z*);

схема десятичной коррекции *СДК*. Позволяет привести результат операции к двоично-десятичному виду (команда *DAA*).

Блок регистров *БРг* предназначен для приема, временного хранения и выдачи информации. *БРг* включает:

счетчик команд *СК* (16-разрядный). Используется для приема и хранения текущего адреса команды. Содержимое *СК* автоматически инкрементируется после выборки каждого байта команды;

регистр указателя стека *УС* (16-разрядный). Служит для приема и хранения ячейки стека, к которой было последнее обращение. Содержимое *УС* инкрементируется, когда данные загружаются в стек, и декрементируется при чтении;

регистры общего назначения *РОН*. Могут быть использованы как накопители данных или указатели адресов операндов. В состав *РОН* входят регистры *В*, *С*, *D*, *Е*, *Н* и *L* (каждый 8-разрядный), которые могут работать в паре, например: *В—С*; *D—Е*; *Н—L*;

регистры временного хранения *W*, *Z* (не адресуемые). Предназначены для приема и временного хранения второго и третьего байтов команд переходов. Образуют регистровую пару *W-Z*;

регистр адреса *RA* (16-разрядный). Служит для приема и хранения в течение одного машинного цикла адреса команды или операнда и выдачи его на буфер *БА* и схему *I/D*. В командах IN и OUT второй байт, дублируясь в регистрах *Н* и *L*, выводится в виде 16-разрядного адреса внешнего устройства. Данные для *RA* коммутируются через мультиплексор *М3* (16-разрядный);

мультиплексоры *М1* и *М2* (8-разрядные). Служат для передачи данных с *ВМД* в блок *РОН* и обратно;

схему инкремента-декремента *I/D*. Представляет собой схему переноса — заема, позволяющую автоматически увеличить или уменьшить на единицу содержимое *RA* и совместить во времени модификацию адреса с выполнением операции в *АЛУ*.

Схема синхронизации *СС* выполняет стандартные функции и состоит из следующих схем формирования:

машинных тактов *СФМТ*. Вырабатывает тактовые импульсы *Т1* ... *Т5*, равные по длительности периоду импульсов синхронизации (в течение одного такта выполняется одна микрооперация);

машинных циклов *СФМЦ*. Используется для выработки цик-

лов М1 ... М5, равных по длительности трем, четырем или пяти тактам;

сигнала *Синхро СФС*. Предназначена для выработки сигнала *Синхро*, определяющего начало каждого МЦ.

Схема управления *СУ* вырабатывает микрооперации, необходимые для выполнения команд в МП. *СУ* включает:

регистр команд *РК* — для приема кода команды (первого байта), поступающего в МП в такте *Т3* цикла М1, и хранения этого кода во время исполнения команды;

программируемые логические матрицы *ПЛМ* — для расшифровки кода команды и выработки микроопераций в соответствии с «зашитой» микропрограммой выполнения команды;

схемы управления *АЛУ* (*СУАЛУ*), регистрами *СУПР1*... *СУПР3*, формирователями машинных тактов *СУМТ* и циклов *СУМЦ*. *СУАЛУ* управляет *R1*, *R2*, *КП* и *СМ*; *СУПР1* ... *СУПР3* предназначены для выработки сигналов управления схемами *А*, *М3*, *СК*, *УС*, *В*, *С*, *D*, *Н*, *L*; *М1*, *М2* соответственно;

схему выборки регистра *СВР*, представляющую собой дешифратор разрядов кода команды, указывающих номер регистра источника либо приемника операндов;

схему анализа переходов *САП*, которая анализирует состояние триггеров регистра условий при выполнении команд условных переходов;

схему выдачи состояния *СВС* — для выдачи 9-разрядного кода состояния на внешний регистр. Позволяет увеличить фактическое число управляющих выводов МП в каждом МЦ за счет временного мультиплексирования информации на шине данных.

Схема *СУОП* состоит из схем анализа прерывания *САПР*, анализа захвата шин *САЗШ* и анализа готовности *САГ*.

САПР обеспечивает возможность работы МП в реальном масштабе времени. *САПР* состоит из триггера разрешения прерывания и триггера запроса прерывания. Анализ наличия запросов прерывания происходит в последнем такте последнего МЦ текущей команды (кроме команд EI и DI), если установлен триггер разрешения прерывания,

САЗШ обеспечивает режим прямого доступа в память внешнего устройства. В этом режиме МП находится до тех пор, пока действует внешний сигнал *Зх*. *САГ* позволяет МП работать с внешними «медленными» устройствами. Используя сигнал *Г*, внешнее устройство может «заставить» ждать МП.

Буфер адреса *БА* представляет собой выходные формирователи с тремя устойчивыми состояниями. *БА* предназначен для выдачи 16-разрядного адреса с *RA* на шину адреса.

Буфер данных *БД* представляет собой бинаправленную трехстабильную схему, применяемую для обмена информацией МП с внешними устройствами. При выводе информации содержимое *ВМД* запоминается в 8-разрядном регистре *БД* и через выходные формирователи выдается на внешнюю *ШД*.

Схема формирования сброса *СФСб* предназначена для трансляции внешнего сигнала *R* на внутрипроцессорные функциональные узлы.

ВМД осуществляет обмен информацией между различными узлами МП. Каждая линия магистрали представляет собой распределенную емкость, влияющую на динамические характеристики МП. Уменьшить время заряда этой емкости в каждом акте обмена позволяет схема заряда магистрали данных *СЗМ*.

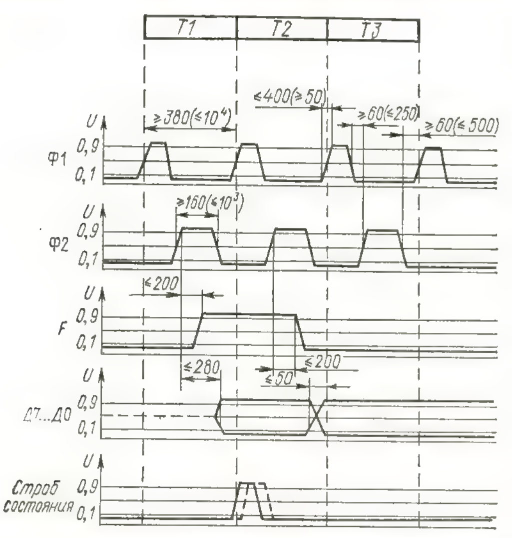


Рис. 3.3. Временная диаграмма выдачи информации состояния

Буфер адреса *БА* состоит из блока адреса *БлАдр* и схемы управления *СУпр*.

Буфер данных *БД* состоит из блока данных *БлД* и схемы управления *СУБД* блоком данных.

Функционирование МП определяется исполняемыми командами. Командный цикл — это время, необходимое для считывания и выполнения очередной команды. Выборка очередной команды осуществляется по содержимому *RA* из *ЗУ* команд. Выработанная команда заносится на *RK* и затем дешифрируется в последовательность микрокоманд. Все команды состоят из одного, двух или трех байтов. В зависимости от типа каждая команда выполняется за время от одного до пяти МЦ, в каждом из которых осуществляется обращение к одной ячейке ЗУ или одному устройству ввода—вывода.

Исключение составляет команда DAD, которая выполняется за три МЦ, но обращение к ЗУ происходит только в первом МЦ. МЦ может длиться три, четыре или пять МТ. Число МТ в машинном цикле определяется кодом выполнения команды. Длительность каждого МТ определяется как интервал между положительными фронтами импульсов синхронизации *Ф1*.

В каждом МЦ микропроцессор вырабатывает сигнал *Синхро* и выдает на ШД информацию состояния в тактах *T1* и *T2* (рис. 3.3). Сигнал *Синхро* обычно характеризует начало очередного МЦ. Сигнал *Синхро* выдается в *T1* по положительному фронту *Ф2* и заканчивается в *T2* по положительному фронту *Ф2*. Для рассмотрения дальнейших действий микропроцессора в МЦ обратимся к диаграмме, приведенной на рис. 3.4. При подаче на схему

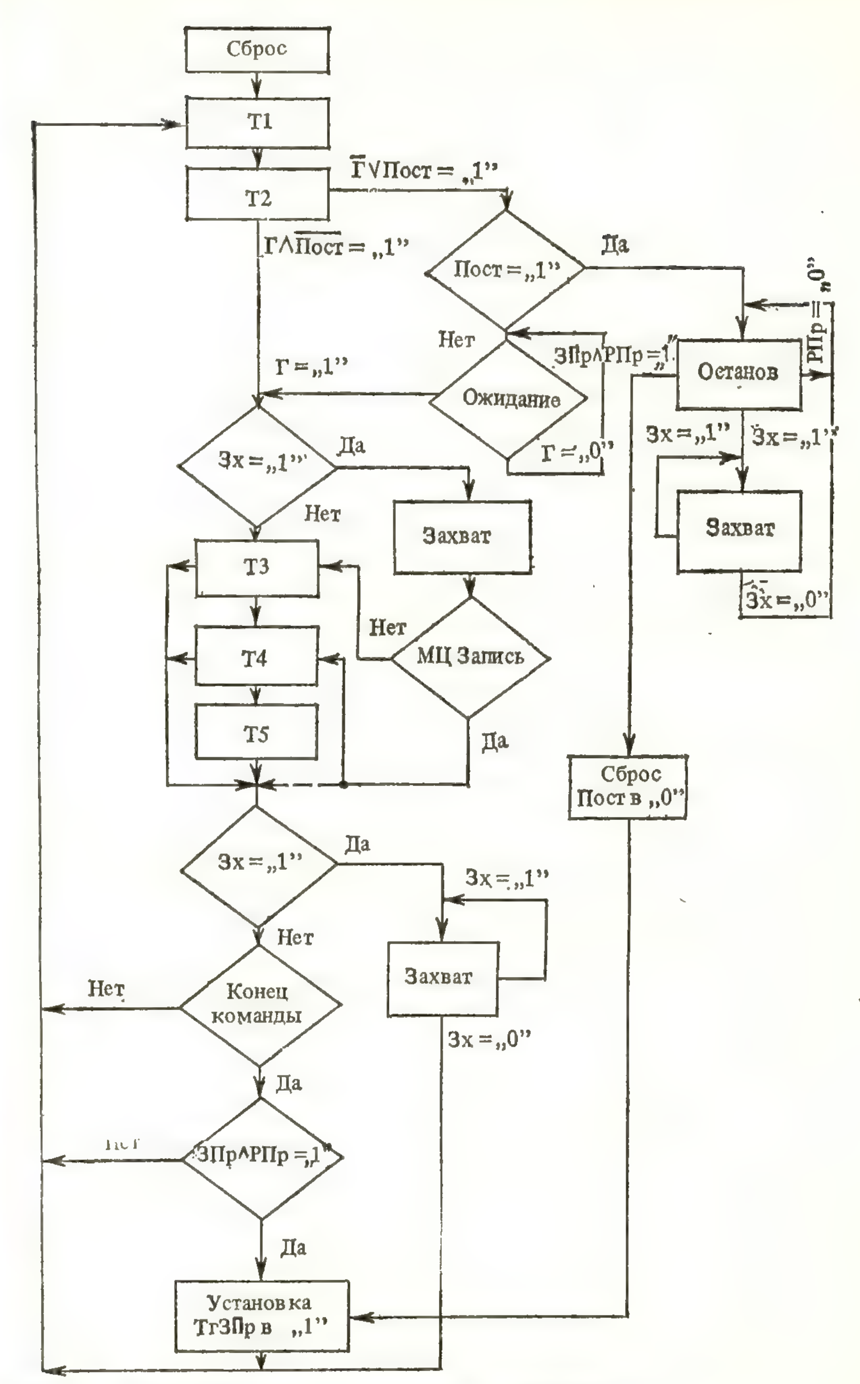


Рис. 3.4. Диаграмма работы микросхемы КР580ИК80А

Таблица 3.3. Действие сигналов состояний

Сигнал состояния	Мнемокод	Разряд шины данных	Действие сигнала
Подтверждение прерывания	ППр	0	Стробирование команды *RST* в МП
Запись — вывод	Зп — Выв	1	Данные выдаются из МП
Стек	Ст	2	На *ША* содержимое регистра *УС*
Подтверждение останова	Пост	3	МП перешел в состояние «Останов»
Вывод	Выв	4	На *ША* адрес устройства; из МП на шину *ШД* выдаются данные
Машинный цикл 1	М1	5	МП принимает код команды
Ввод	Вв	6	На *ША* адрес устройства; МП принимает данные по *ШД*
Чтение	Чт	7	МП читает содержимое ЗУ

сигнала *Сброс* МЦ начинается с такта *Т1*. В этом МЦ происходят обращение к ЗУ и выдача информации состояния в виде параллельного 8-разрядного кода. Назначение и состояние каждого разряда приведены в табл. 3.3. В табл. 3.4 дана дополнительная информация о сигналах состояния.

Из машинного такта *Т1* МП всегда переходит к такту *Т2*. В этом такте анализируются сигналы *Г*, *ЗхВ* и сигнал состояния «Подтверждение останова». Если последний в состоянии «1» или сигнал *Г* = «0», то МП переходит в состояние «Останов» (Тост.) или «Ожидание» (Тож.) соответственно. В состоянии «Ожидание» МП будет находиться до тех пор, пока не поступит сигнал *Г* = «1». Из состояния «Останов» МП может выйти в случае поступления следующих сигналов: *Зх* = «1», *Зпр* = «1» и при взведенном внутреннем триггере «Разрешение прерывания» *R* = «1».

Таблица 3.4. Распределение сигналов состояния по машинным циклам

МЦ	Сигналы состояния (разряды шины данных)							
	ППр (0)	Зп Выв (1)	Ст (2)	Пост (3)	Выв (4)	М1 (5)	Вв (6)	Чт (7)
М1	1 [8]	—	—	1 [4]	—	1	—	1 [3]
М2	—	—	—	—	—	—	—	1 [3]
М3	—	—	—	—	—	—	—	1 [3]
М4	—	1 [7]	1 [6]	1 [5]	—	—	—	1 [1] *
М5	—	1 [7]	—	—	1 [8]	—	1 [2]	—

* При выполнении команд: 1 — требующих чтение памяти; 2 — IN; 3 — OUT; 4 — при выходе из режима «Останов по запросу прерывания»; 5 — при выполнении команд — HALT; 6 — со стеком; 7 — OUT и команд, требующих записи в память; 8 — при поступлении сигнала Зпр.

Если на входе *Г* напряжение «1», то МП переходит к анализу состояния сигнала *Зх*. При *Зх* = «1» МП переходит в режим «Захват», тем самым предоставляя *ША* и *ШД* внешнему, активному устройству. МП вырабатывает сигнал *ПЗхв* и направляет его внешнему устройству. Если текущий МЦ не является циклом «Запись», то сигнал *ПЗхв* выдается по положительному фронту *Ф1* в такте, следующим за *Т3*. После освобождения системных шин внешнее устройство сбрасывает сигнал *Зх* (*Зх* = «0») и МП начинает выполнять следующий машинный цикл прерываний команды или первый машинный цикл новой команды с такта *Т1*. На рис. 3.5 и 3.6 приведены временны́е диаграммы, характеризующие состояние сигнала *Зх* в машинных циклах «Чтение памяти» и «Запись в память». Сигнал *Зх* необходимо устанавливать не ранее положительного фронта *Ф2*. Это может быть выполнено синхронизацией сигнала *Зх* отрицательным фронтом *Ф1*. Сигнал *ПЗх* выдается по положительному фронту *Ф1* в *Т3*, в то же время *ШД* переходит в третье устойчивое состояние. *ША* переходит в третье состояние в такте, следующем за *Т3* (*Т4* либо первый такт состояния «Захват»). Сброс сигнала *Зх* можно синхронизировать отрицательным фронтом *Ф2*.

После выполнения *Т3 МП* может перейти к *Т4*, либо закончить текущий МЦ (в зависимости от типа МЦ). Это же относится и к *Т4*.

В течение *Т4* и *Т5* МП выполняет внутренние операции. В это время не происходит обращения к внешним устройствам.

В конце МЦ вновь анализируется сигнал *Зх*. Если этот сигнал соответствует уровню «1», то выполнение действий режима «Захват» продолжается. В противном случае МП заканчивает прерванную команду. После последнего МЦ анализируется сигнал *Зх*. При наличии на соответствующем входе «1» и при условии, что внутренний триггер «Разрешения прерывания» также в состоянии «1», внутренний триггер «Запроса прерывания» установится «1», МП выдает сигнал *РПр*. Внутренний триггер *Разрешения прерывания* взводится командой EI и сбрасывается командой DI.

При отсутствии запросов прерывания МП переходит к выполнению следующей команды.

В табл. 3.5 даны типы машинных циклов.

Первым МЦ каждой команды является *Выбор кода команды* (рис. 3.7). В этом цикле содержимое *СК* выдается на *ША* по положительному фронту *Ф2* в Т1. Эта информация остается неизменной до положительного фронта *Ф2* такта, следующего за Т3. Код команды принимается по *ШД* на *RK* в такте Т3. Декодирование кода происходит в Т4 и Т5. Сигнал может быть использован для нормального взаимодействия МП с медленными ЗУ. МЦ «Чтение памяти» (рис. 3.8) отличается от цикла «Выборка кода команды» тем, что отсутствует сигнал состояния М1, и байт информации заносится из внутренних *РОН*.

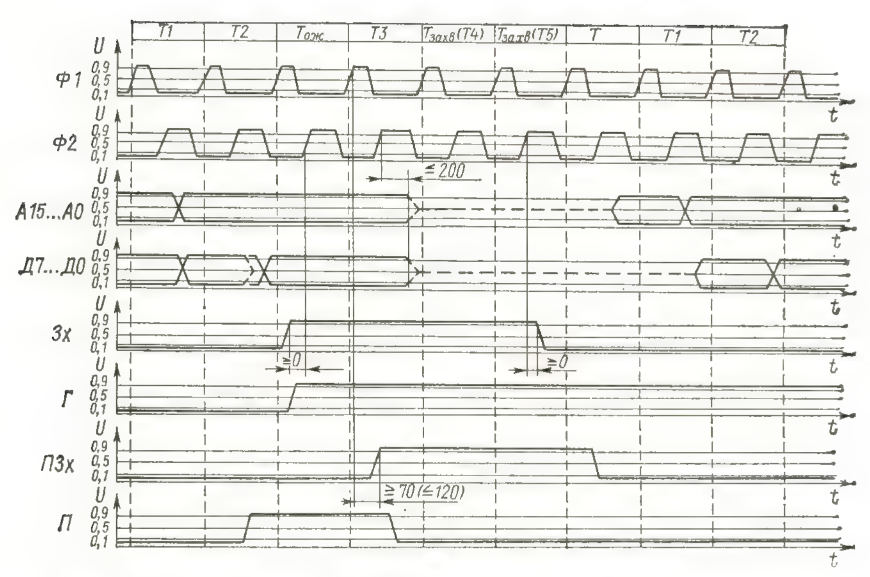


Рис. 3.5. Состояние «Захват» в МЦ «Чтение памяти»

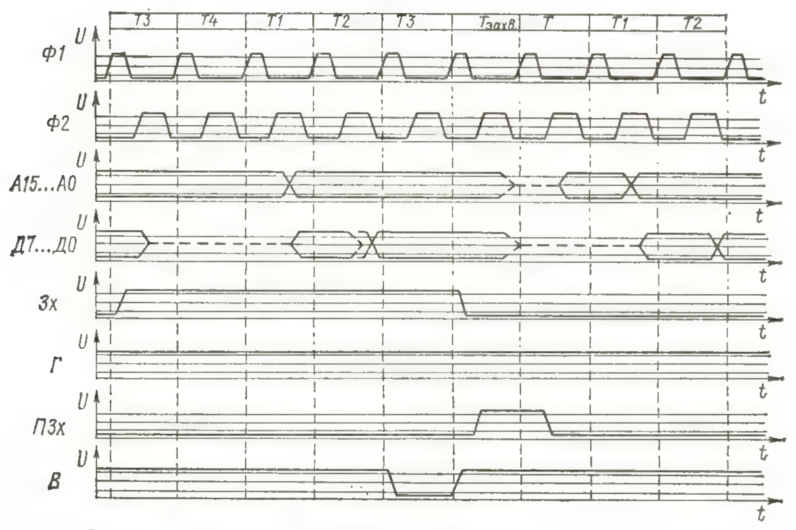


Рис. 3.6. Состояние «Захват» в МЦ «Запись в память»

Рис. 3.7. МЦ «Выборка кода команды»

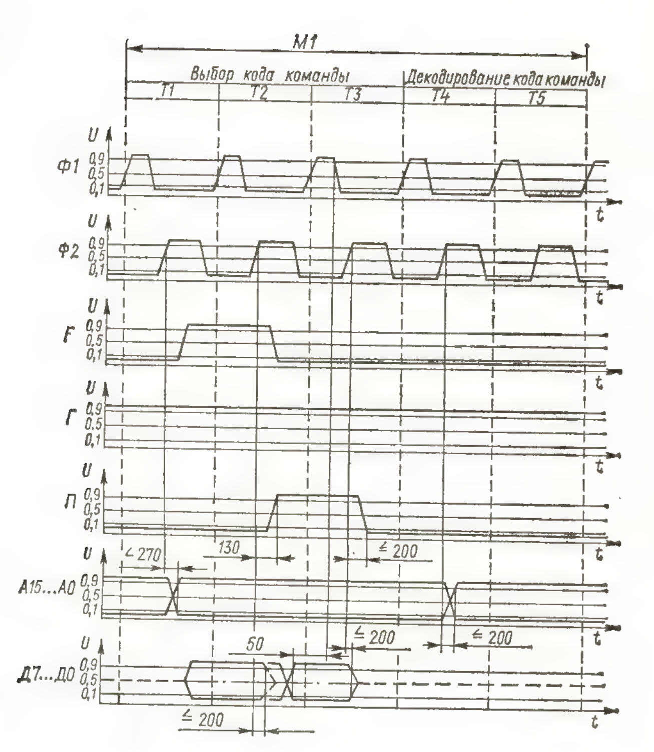


Рис. 3.8. МЦ «Чтение памяти»

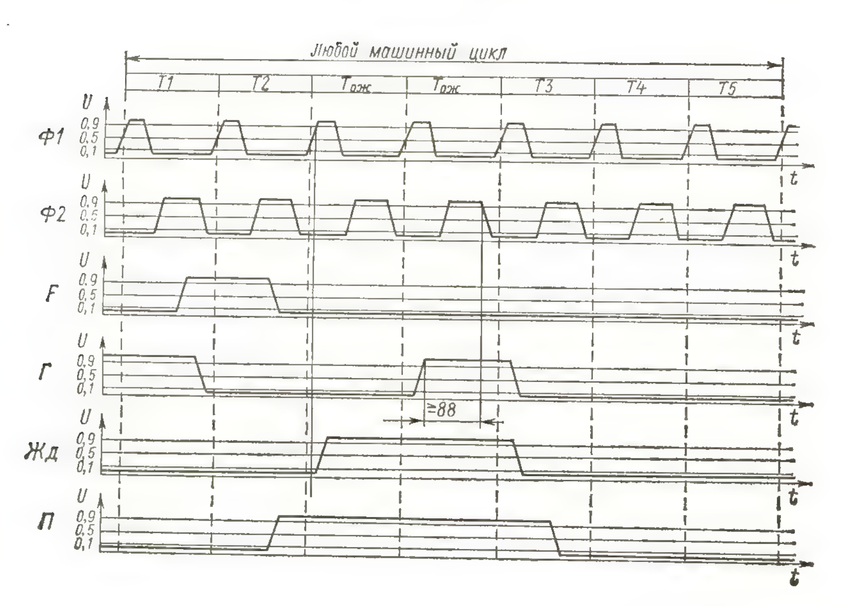


Таблица 3.5. Сигналы состояния, формируемые в машинных циклах

Тип МЦ	Сигналы состояния							
	ППр	Зп-Выв	Ст	Пост	Выв	М1	Вв	Чт
Выборка кода команды	0	0	0	0	0	1	0	1
Чтение памяти	0	0	0	0	0	0	0	1
Запись памяти	0	1	0	0	0	0	0	0
Чтение стека	0	0	1	0	0	0	0	1
Запись в стек	0	1	1	0	0	0	0	0
Чтение устройства	0	0	0	0	0	0	1	0
Запись в устройство	0	1	0	0	1	0	0	0
Подтверждение прерывания	1	0	0	0	0	1	0	0
Подтверждение останова	0	0	0	1	0	0	0	1
Подтверждение прерывания при останове	1	0	0	1	0	1	0	0

МЦ «Запись в память» (рис. 3.9) отличается от МЦ «Чтение памяти» тем, что данные из МП через шину данных записываются в ячейку, адрес которой поступает по *ША*.

Этот машинный цикл обычно бывает четвертым или пятым циклом исполнения команды. В МЦ «Запись в память» данные на *ША* выводятся из МП в Т1 и снимаются по положительному фронту *Ф2* в такте Т2. МЦ «Запись в стек» и «Чтение стека» идентичны циклам «Запись в память» и «Чтение памяти» за исключением дополнительного сигнала состояния *Ст*. В частности, этот сигнал может быть использован для запрета выборки «нестековой» части памяти при упомянутых циклах.

МЦ «Чтение устройства» и «Запись в устройство» могут наступать только при выполнении команд IN и OUT соответственно и отличаются от МЦ «Чтение памяти» и «Запись в память» наличием в информации состояния сигналов Вв и Выв соответственно.

МЦ «Подтверждение прерывания» (рис. 3.10) во многом аналогичен МЦ «Выбор кода команды», невозможно лишь изменение содержимого *СК*. Устройство, вызвавшее прерывание, формирует команду (обычно RST) и выдает ее на *ШД* в *Т3*.

МЦ «Подтверждение останова» (рис. 3.11) наступает в результате выполнения команды HLT и характеризуется сигналами состояния *Чт* и *Пост*.

МЦ «Подтверждение прерывания при останове» аналогичен МЦ «Подтверждение прерывания». Единственное отличие — появление сигнала состояния *Пост* по положительному фронту *Ф2* в Т1.

Во время работы МП может перейти в машинные состояния «Ожидание», «Захват» и «Останов», длительность которых определяется внешними сигналами *Г*, *Зх*, *ЗПр* (см. рис. 3.12 и 3.13).

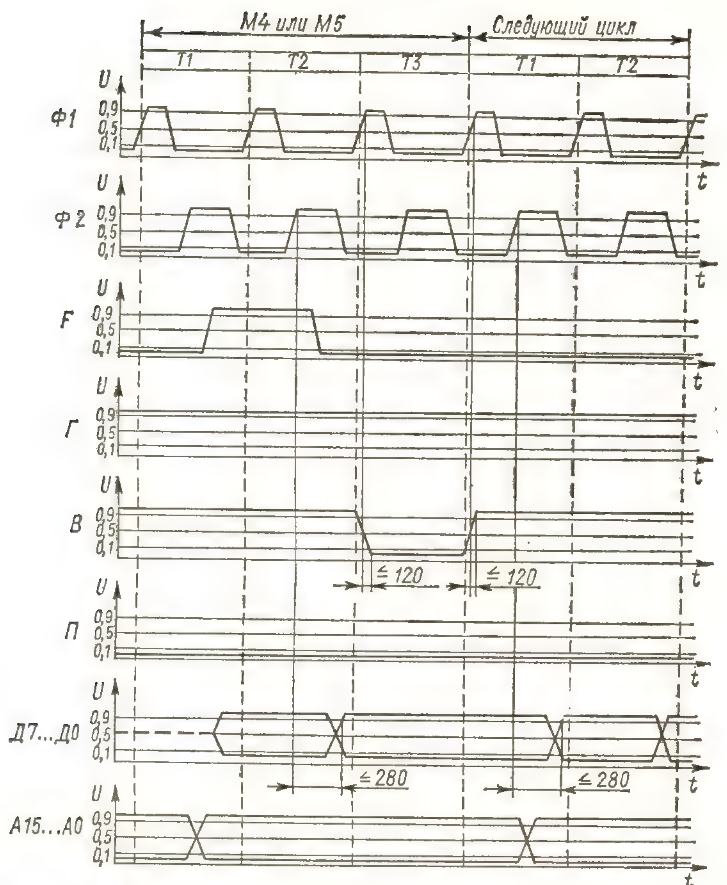


Рис. 3.9. МЦ «Запись в память»

Состояние «Ожидание» используется для введения задержки между Т2 и Т3 при обращении к медленно действующим внешним устройствам.

В состоянии «Захват» ШД и ША МП переводятся в третье состояние. Это позволяет организовать несколько активных устройств на системной шине. На рис. 3.12 и 3.13 показана работа МП в состоянии «Останов».

Следует остановиться на порядке включения и отключения источников питания. Наиболее благоприятный режим коммутации питания — автоматическое одновременное отключение и включение трех источников питания. Допускается неодновременная подача питающих напряжений в следующей последовательности: U_{CC} *3*, U_{CC} *2*, U_{CC} *1*.

Отключение производится в обратном порядке.

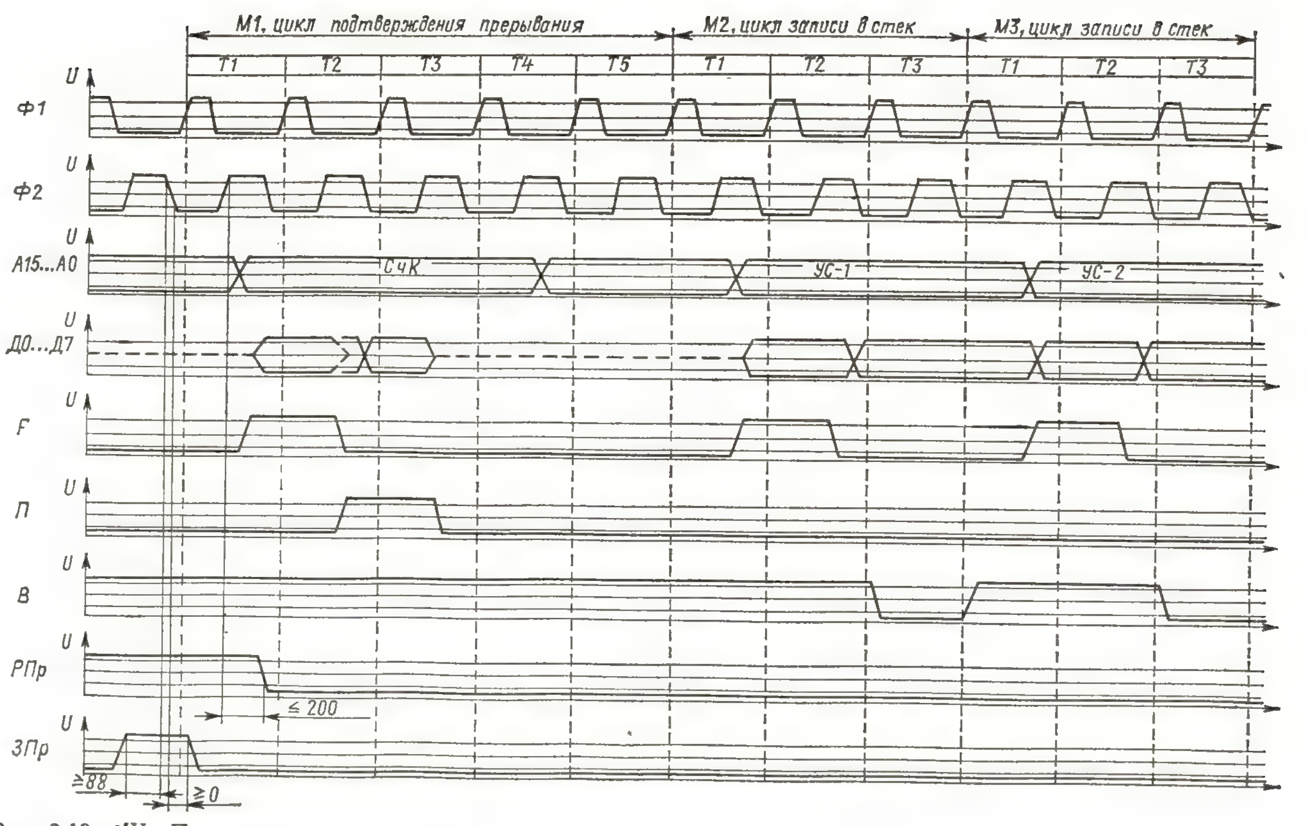


Рис. 3.10. МЦ «Подтверждение прерывания»

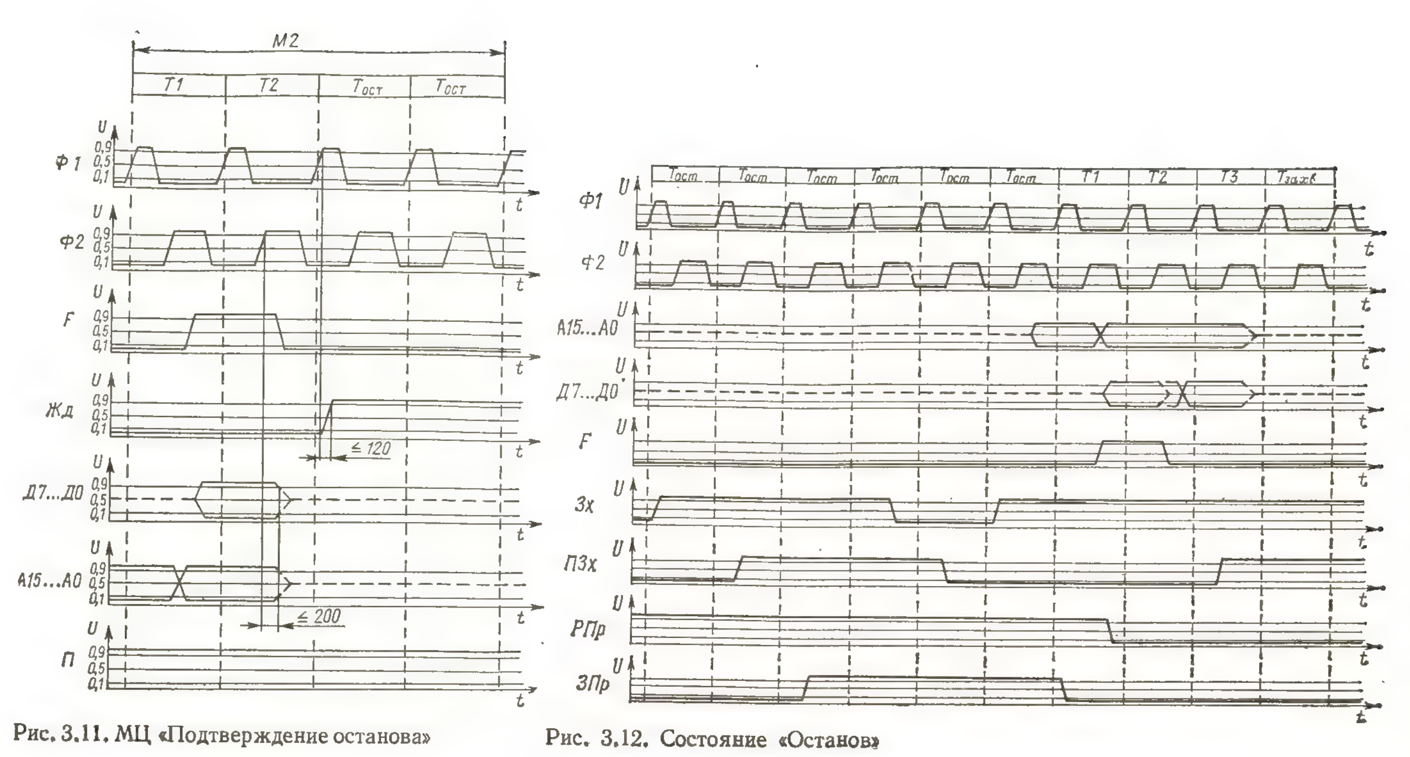


Рис. 3.11. МЦ «Подтверждение останова»

Рис. 3.12. Состояние «Останов»

Таблица 3.6. Действия МП при выполнении команд, продолжительностью до трех

Команда	МЦ1 (п. 1)		МЦ2	
	Т4	Т5	Т1	Т2 (п. 6)
MOVR_iR_k	(R_k) → Р1 (п. 3)	(R1) → R_i (п. 3)	—	—
MOVR_iM			(H, L) → ША, ВС (п. 4)	ШД → R_i
MOVM, R_k	(R_k) → R1	—	(H, L) → ША, ВС (п. 5)	R1 → ШД
SPHL	(H, L) → УС	(H, L) → УС	—	—
MVIP	—	—	(СК) → ША	⟨B2⟩ → R
MVIM	—	—	То же	⟨B2⟩ → R1
LDAX	—	—	(RP) → ША, ВС (п. 4)	ШД → A
STAX	—	—	(RP) → ША,	(A) → ШД
XCHL	(H, L) → (D, E)	—	—	—
ADDR	(R) → R1, (A) → R2	—	(п. 8)	(R1) + (R2) → → A
ADDM	(A) → R2	—	(H, L) → ША, ВС (п. 4)	(ШД) → R1
ADI	(A) → R2	—	(СК) → ША, ВС (п. 4)	(СК) + 1 → → СК, ⟨B2⟩ → R1
ADCR	(R) → R1 (A) → R2	—	(п. 8)	(R1) + (R2) + + (C) → A
ADCM	(A) → R2	—	(H, L) → ША, ВС (п. 4)	ШД → R1
ACI	(A) → R2	—	(СК) → ША, ВС (п. 4)	⟨B2⟩ → R1 (СК) + 1 → СК
SUBR	(R) → R1, (A) → R2	—	(п. 8)	(R2) — (R1) → → A
SUBM	(A) → R2	—	(H, L) → ША, ВС (п. 4)	ШД → R1
SUI	(A) → R2	—	(СК) → ША, ВС (п. 4)	(СК) + 1 → → СК, ⟨B2⟩ → R1
SBBR	(R) → R1, (A) → R2	—	(п. 8)	(R2) — (R1) — — (C) → A
SBBM	(A) → R2	—	(H, L) → ША, ВС (п. 4)	ШД → R1
SBI	(A) → R2	—	(СК) → ША, ВС (п. 4)	(СК) + 1 → СК, ⟨B2⟩ → R1
INRR	(R) → R1, (R1) + 1 → → АЛУ	(АЛУ) → R	—	—
INRM	—	—	(H, L) → ША, ВС (п. 4)	ШД → R1, (R1) + 1 → → АЛУ
DCRR	(R) → R1 (R1) — 1 → → АЛУ	(АЛУ) → R	—	—
DCRM	—	—	(H, L) → ША, ВС (п. 4)	ШД → R1

машинных циклов

	МЦ3			Примечания
Т3	Т1	Т2 (п. 6)	Т3	
—	—	—	—	(п. 2)
ШД → R_i	—	—	—	—
$R1$ → ШД	—	—	—	—
—	—	—	—	—
⟨$B2$⟩ → R	—	—	—	(п. 7)
⟨$B2$⟩ → $R1$	(СК) → ША ВС (п. 4)	($R1$) → ШД	($R1$) → ШД	
ШД → A	—	—	—	RP — пара регистров
(A) → ШД	—	—	—	—
—	—	—	—	—
—	—	—	—	(п. 7)
(ШД) → $R1$	(п. 8)	($R1$) + ($R2$) → → A	—	—
⟨$B2$⟩ → $R1$	(п. 8)	($R1$) + ($R2$) → → A	—	—
—	—	—	—	(п. 7)
ШД → $R1$	(п. 8)	($R1$) + ($R2$) + + (C) → A	—	—
⟨$B2$⟩ — $R1$	(п. 8)	То же	—	
—	—	—	—	(п. 7)
ШД → $R1$	(п. 8)	($R2$) — ($R1$) → → A	—	—
⟨$B2$⟩ → $R1$	(п. 8)	($R2$) — ($R1$) → → A	—	—
—	—	—	—	(п. 7)
ШД → $R1$	(п. 8)	($R2$) — ($R1$) — — (C) → A	—	—
⟨$B2$⟩ → $R1$	(п. 8)	То же	—	—
—	—	—	—	(п. 7)
ШД → $R1$, ($R1$) + 1 → → АЛУ	(H, L) → ША, ВС (п. 5)	(АЛУ) → ШД	—	—
—	—	—	—	(п. 7)
ШД → $R1$, ($R1$) — 1 → → АЛУ	(H, L) → ША, ВС (п. 5)	(АЛУ) → ШД	(АЛУ) → ШД	—

Команда	МЦ1 (п. 1)		МЦ2	
	T4	T5	T1	T2 (п. 6)
INXRP	$(RP)+1 \to RP$	(п. 9)	—	—
DCXRP	$(RP)-1 \to RP$	(п. 9)	—	—
DAA	$(A) \to A$, (п. 10)	—	—	—
ANAR	$(R) \to R1$, $(A) \to R2$	—	(п. 8)	$(R1) \wedge (R2) \to A$
ANAM	$(A) \to R2$	—	$(H, L) \to$ ША, ВС (п. 4)	ШД $\to R1$
ANI	$(A) \to R2$	—	(СК) $\to$ ША, ВС (п. 4)	(СК) $+ 1 \to$ СК, $\langle B2 \rangle \to R1$
XRAR	$(R) \to R1$, $(A) \to R2$	—	(п. 8)	$(R1) \oplus (R2) \to A$
XRAM	$(A) \to R2$	—	$(H, L) \to$ ША, ВС (п. 4)	ШД $\to R1$
XRI	$(A) \to R2$	—	(СК) $\to$ ША, ВС (п. 4)	(СК) $+ 1 \to$ СК $\langle B2 \rangle \to R1$
ORAR	$(A) \to R2$, $(R) \to R1$	—	(п. 8)	$(R2) \vee (R1) \to A$
ORAM	$(A) \to R2$	—	$(H, L) \to$ ША, ВС (п. 4)	ШД $\to R1$,
ORI	$(A) \to R2$	—	(СК) $\to$ ША, ВС (п. 4)	(СК) $+ 1 \to$ СК $\langle B2 \rangle \to R1$
CMPR	$(A) \to R2$, $(R) \to R1$	—	(п. 8)	$(R2) - (R1)$
CMPM	$(A) \to R2$	—	$(H, L) \to$ ША, ВС (п. 4)	ШД $\to R1$
CPI	$(A) \to R2$	—	(СК) $\to$ ША, ВС (п. 4)	(СК) $+ 1 \to$ СК, $\langle B2 \rangle \to R1$
RLC	$(A) \to$ АЛУ, сдвиг левый	—	(п. 8)	(АЛУ) $\to A, C$
RRC	$(A) \to$ АЛУ, сдвиг правый	—	(п. 8)	То же
RAL	$(A), C \to$ АЛУ, Сдвиг левый	—	(п. 8)	»
RAR	$(A), C \to$ АЛУ, сдвиг правый	—	(п. 8)	(АЛУ) $\to A, C$
CMA	$(\bar{A}) \to A$	—	—	—
CMC	$\bar{C} \to C$	—	—	—
STC	$1 \to C$	—	—	—
PCHL	$(H, L) \to$ СК	(п. 9)	—	—
PUSHRP (п. 7)	(УС) $- 1 \to$ УС	(п. 9)	(УС) $\to$ ША, ВС (п. 12)	(УС) $- 1 \to$ УС, $(R_{ст}) \to$ ШД
PUSHPSW	(УС) $- 1 \to$ УС	(п. 9)	(УС) $>$ ША, ВС (п. 12)	(УС) $- 1 \to$ УС, $A \to$ ШД
POPRP	—	—	(УС) $\to$ ША, ВС (п. 13)	(УС) $+ 1 \to$ УС, ШД $\to R_{мл}$
POPPSW	—	—	(УС) $\to$ ША, ВС (п. 13)	(УС) $+ 1 -$ УС, ШД $\to F$
EI	$1 \to$ ТгРПР	—	—	—
DI	$0 \to$ ТгРПР	—	—	—
NOP	—	—	—	—

Продолжение табл. 3.6

	МЦ3			Примечания
Т3	Т1	Т2 (п. 6)	Т3	
—	—	—	—	—
—	—	—	—	—
—	—	—	—	—
—	—	—	—	(п. 7)
ШД → $R1$	(п. 8)	$(R1) \wedge (R2) \to A$	—	—
$\langle B2 \rangle \to R1$	(п. 8)	$(R1) \wedge (R2) \to A$	—	—
—	—	—	—	(п. 7)
ШД → $R1$	(п. 8)	$(R1) \oplus (R2) \to A$	—	—
$\langle B2 \rangle \to R1$	(п. 8)	$(R1) \oplus (R2) \to A$	—	—
—	—	—	—	(п. 7)
ШД → $R1$	(п. 8)	$(R1) \vee (R2) \to A$	—	—
$\langle B2 \rangle \to R1$	(п. 8)	$(R1) \vee (R2) \to A$	—	—
—	—	—	—	(пп. 7, 10)
ШД → $R1$	(п. 8)	$(R2) - (R1)$	—	(п. 10)
$\langle B2 \rangle \to R1$	(п. 8)	$(R2) - (R1)$	—	(п. 10)
—	—	—	—	(п. 11)
—	—	—	—	—
—	—	—	—	—
—	—	—	—	—
—	—	—	—	—
—	—	—	—	(п. 11)
—	—	—	—	(п. 11)
—	—	—	—	—
$(R_{ст})$ — ШД	ВС (п. 12)	$(R_{мл})$ → ШД	$(R_{мл})$ → ШД	—
A → ШД	(УС) → ША, ВС (п. 12)	(F) → ШД	(F) → ШД	F — регистр признаков
ШД → $R_{мл}$	(УС) → ШД, ВС (п. 13)	(УС) + 1 → УС, ШД → $R_{ст}$	ШД → $R_{ст}$	—
ШД → F	(УС) → ША, ВС (п. 13)	(УС) + 1 → УС ШД → A	ШД → A	—
—	—	—	—	—
—	—	—	—	—
—	—	—	—	(п. 10)

Команда	МЦ1 (п. 1)		МЦ2	
	T4	T5	T1	T2 (п. 6)
LXIRP	—	—	(СК) → ША, ВС (п. 4)	(СК) + 1 → СК, ⟨B2⟩ → $R_{мл}$
DADRP	—	—	($R_{мл}$) → R2	(L) → R1, (R2) + (R1) → → АЛУ
JMP	—	—	(СК) → ША, ВС (п. 4)	(СК) + 1 → СК, ⟨B2⟩ → R
J (условие)	Проверка условия	Проверка условия	(СК) → ША, ВС (п. 4)	(СК) + 1 → СК, ⟨B2⟩ → R
R (условие)	То же	То же (п. 16)	(УС) → ША, ВС	(УС) + 1 — УС, ШД → Z
RST	(УС) — 1 → УС	(п. 19)	То же	(УС) — 1 → УС, ($R_{ст}$) → ШД
RET	—	—	(УС) → ША, ВС (п. 13)	(УС) + 1 → УС, ШД → Z

Таблица 3.7. **Действия МП при выполнении продолжительных команд**

Команда	МЦ1		МЦ2			МЦ3	
	T4	15	T1	T2	T3	T1	T2
LDA	—	—	(СК) → ША, ВС (п. 14)	(СК) + 1 → → СК, ⟨B2⟩ → Z	⟨B2⟩ → → Z	(СК) → ША, ВС (п. 4)	(СК) + 1 → → СК, ⟨B3⟩ → W
STA	—	—	То же	То же	То же	То же	То же
LHLD	—	—	»	»	»	»	»
SHLD	—	—	»	»	»	»	»
CALL	—	(п. 9)	»	»	»	»	»
C (условие) п. 15	Проверка условия: «Да» — (УС) —1 → → УС	(п. 9)	»	»	»	»	»

	МЦ3			Примечания
Т3	Т1	Т2 (п. 6)	Т3	
⟨B2⟩ → $R_{мл}$	(СК) → ША, ВС (п. 4)	(СК) + 1 → → СК, B3 → $R_{ст}$	⟨B3⟩ → $R_{ст}$	—
—	($R_{ст}$) → R2	(H) → R1, (R2) + (R1) + + C → АЛУ	(АЛУ) → H, L	—
⟨B2⟩ → R	(СК) → ША, ВС (п. 4)	(СК) + 1 → → СК, <B3> → R	⟨B3⟩ → R	(п. 15)
⟨B2⟩ → R	(СК) → ША, ВС (п. 4)	(СК) + 1 → → СК, <B3> → R	⟨B3⟩ → R	(п. 15)
ШД → Z	(УС) — ША, ВС (п. 12)	(УС) + 1 → → УС, ШД → R	ШД → W	
($R_{ст}$) → ШД	(УС) → ША, ВС (п. 12)	$R_{мл}$ → ШД	$R_{мл}$ → ШД	
ШД → Z	(УС) → ША, ВС (п. 13)	(УС) + 1 → → УС, ШД → W	ШД → W	(п. 15)

	МЦ4			МЦ5		
Т3	Т1	Т2	Т3	Т1	Т2	Т3
⟨B3⟩ → → W	(W, Z) → → ША, ВС (п. 4)	ШД → A	ШД → A	—	—	—
То же	(W, Z) → → ША, ВС (п. 5)	(A) → ШД	(A) → ШД	—	—	—
»	(W, Z) → → ША, ВС (п. 4)	(W, Z) + 1 → → W, Z, ШД → L	ШД → L	(W, Z) → → ША, ВС (п. 4)	ШД → H	ШД → H
»	(W, Z) → → ША, ВС (п. 5)	(W, Z) + 1 → → W, Z (L) → ШД	(L) → ШД	(W, Z) → → ША, ВС (п. 5)	(H) → ШД	(H) → ШД
»	(УС) → ША, ВС (п. 12)	(УС)−1−УС, $(СК)_{ст}$ →ШД	$(СК)_{ст}$ → →ШД	(УС) → ША, ВС (п. 12)	$(СК)_{мл}$ → → ШД	$(СК)_{мл}$ → → ШД
»	То же	То же	То же	То же	То же	То же

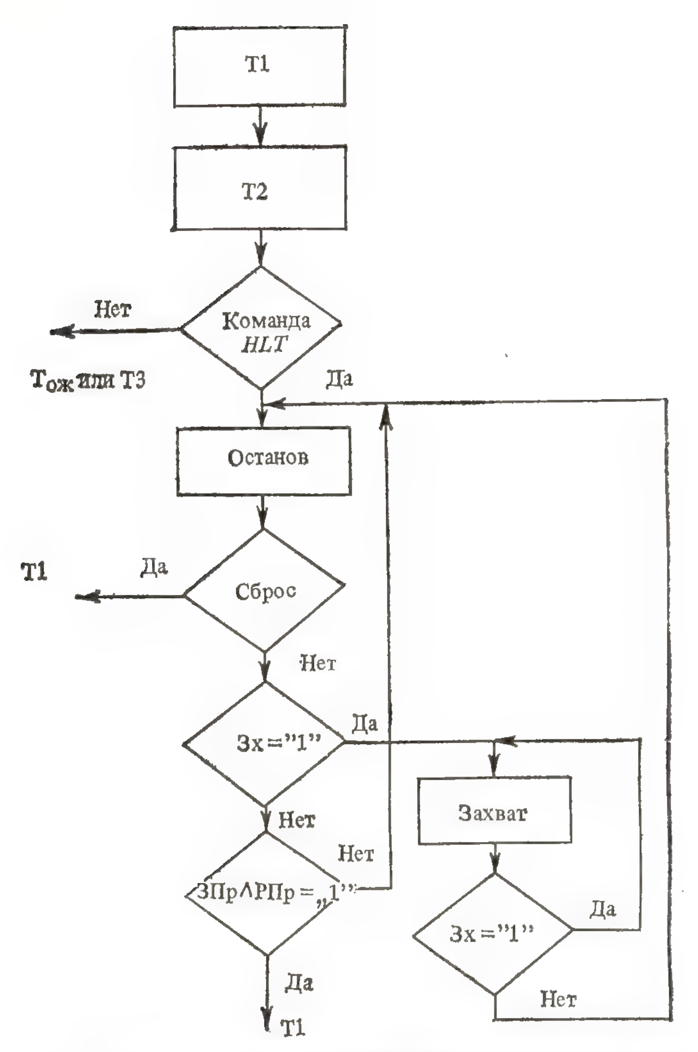


Рис. 3.13. Вывод микросхемы КР580ИК80А из состояния «Останов»

При разработке МП были использованы принципы временного хранения информации на паразитных емкостях, что требует применения фазовых импульсов *Ф2* и *Ф1* строго определенной частоты.

Для более полного понимания правил выполнения каждой команды в табл. 3.6 и 3.7 даны действия МП в каждом машинном такте.

Рассмотрим, например, как выполняется команда $MOVR_iR_j$. В машинном цикле МЦ1 в первом его такте содержимое

Таблица 3.8. Статические параметры микросхемы КР580ИК80А

Параметр, единица измерения	Обозначение	Значения			Режим измерения
		мин.	ном.	макс.	
Напряжение «1» для Ф1 и Ф2, В	$U_{IH\Phi}$	8,5	12,0	13,2	—
Напряжение «0» для Ф1 и Ф2, В	$U_{IL\Phi}$	—1,0	0,45	0,8	—
Входное напряжение «1», В	U_{IH}	3,0	3,7	6,0	—
Входное напряжение «0», В	U_{IL}	1,0	0,45	0,8	—
Выходное напряжение «1», В	U_{OH}	3,7	—	—	I_{OL} = 15 мкА
Выходное напряжение «0», В	U_{OL}	—	—	0,45	I_{OH} = 1,9 мА
Ток потребления от U_{CC1}, мА	I_{CC1}	—	55	85	Длительность цикла 480 нс
Ток потребления от U_{CC2}, мА	I_{CC2}	—	70	110	
Ток потребления от U_{CC3}, мА	I_{CC3}	—	0,1	10	
Выходной ток в состоянии «Отключено», мкА	I_{OZ}	—100	—	100	—
Ток утечки на входах, мкА	I_{IZ}	—10	—	10	—
Входной ток по ШД, мА	I_{ID}	—2,0	—	—	—

счетчика команд выдается на *ША*, информация состояния выводится на *ШД*. Во время действия Т2 содержимое счетчика команд инкрементируется. В Т3 команда с шины данных заносится во внутренние регистры R_j, *PK*. В Т4 содержимое R_j пересылается в R_1. В Т5 содержимое R_1 пересылается в R_i. Выполнение команды оканчивается.

Примечания к табл. 3.6 и 3.7:

п. 1. Во всех командах, кроме RST, во время действия Т1 выдается содержимое *СК* на *ША*, информация состояния выдается на *ШД* [(*СК*) → *ША*, состояние → *ШД*]; во время действия Т2 — (*СК*) + 1 → *СК*; во время действия Т3 — команда (*ШД*) → *PK*, R_1; в Т3 исполнения RST 0 → *W*, команда → *PK*, R_1;

п. 2. i и $k \neq$ M; если k = M, смотри команду MOVR$_i$ M; если i = M, смотри команду MOVM, R$_i$;

п. 3. Такты Т4 и Т5 в этой команде используются для проведения внутренних операций в МП;

п. 4. МЦ «Чтение памяти»;

п. 5. МЦ «Запись в память»;

п. 6. В каждом МЦ в Т2 МП ожидается сигнал *Г*, при его отсутствии МП переходит в режим «Ожидание»;

п. 7. При $R = M$ либо $RP = PSW$ смотри следующую команду;

п. 8. Аккумулятор не загружается результатом, пока не наступит Т2;

п. 9. Функционально этот такт совмещен с предыдущим;

Таблица 3.9. Команды микропроцессора

Команда	Мнемоника	Код			Признаки					Содержание команды
					C	Z	S	P	C'	
Пересылка	MOVR_iR_j	1	R_i	R_j	—	—	—	—	—	$(R_j) \to R_i$
Инкремент	INRR	0	R	4	—	+	+	+	+	$(R) + 1 \to R$
Декремент	DCRR	0	R	5	—	+	+	+	+	$(R) - 1 \to R$
Сложение	ADDR	2	0	R	+	+	+	+	+	$(A) + (R) \to A$
Сложение с переносом	ADCR	2	1	R	+	+	+	+	+	$(A) + (R) + C \to A$
Вычитание	SUBR	2	2	R	+	+	+	+	+	$(A) - (R) \to A$
Вычитание с заемом	SBBR	2	3	R	+	+	+	+	+	$(A) - (R) - C \to A$
Логическое умножение	ANAR	2	4	R	0	+	+	+	—	$(A) \wedge (R) \to A$
Неравнозначность	XRAR	2	5	R	0	+	+	+	—	$(A) \oplus (M) \to A$
Логическое сложение	ORAR	2	6	R	0	+	+	+	—	$(A) \vee (R) \to A$
Сравнение	CMPR	2	7	R	+	+	+	+	+	Операнды (A) и (R): если $(A) = (R)$, то $(Z) = 1$; если $(A) \neq (R)$, то $(C) = 1$
Сдвиг влево с переносом	RAL	0	2	7	+	—	—	—	—	$2A \to A$; $(C) \to A0$; $(A7) \to C$
Сдвиг вправо с переносом	RAR	0	3	7	+	—	—	—	—	$A : 2 \to A$; $(C) \to A7$; $(A0) \to C$
Сдвиг влево циклический	RLC	0	0	7	+	—	—	—	—	$2A \to A$; $(A7) \to A0$; $(A7) \to C$
Сдвиг вправо	RRC	0	1	7	+	—	—	—	—	$A : 2 \to A$; $(A0) \to (A7)$
Загрузка A содержимым ячейки памяти с адресом (D), (E) либо (B), (C)	LDAXD, B	0 (0	3 1	2 2)	—	—	—	—	—	$[(D, C)] \to A$, либо $[(B, C)] \to A$
Инкремент регистровой пары B, D, H, SP	INXB	0	0	3	—	—	—	—	—	$(B, C) + 1 \to B, C$
	INXD	0	2	3	—	—	—	—	—	$(D, E) + 1 \to D, E$
	INXH	0	4	3	—	—	—	—	—	$(H, L) + 1 \to H, L$
	INXSP	0	6	3	—	—	—	—	—	(УС) + 1 → УС
Декремент регистровой пары B, D, H, SP	DCXB	0	1	3	—	—	—	—	—	$(B, C) - 1 \to B, C$
	DCXD	0	3	3	—	—	—	—	—	$(D, E) - 1 \to D, E$
	DCXH	0	5	3	—	—	—	—	—	$(H, L) - 1 \to H, L$
	DCXSP	0	7	3	—	—	—	—	—	(УС) − 1 → УС
Запомнить (A) по адресу (B, C), (D, E)	STAXB	0	0	2	—	—	—	—	—	$A \to (B, C)$
	STAXD	0	2	2	—	—	—	—	—	$A \to (D, E)$
Обмен содержимым регистровых пар H и D	XCHL	3	5	3	—	—	—	—	—	$(H) \leftrightarrow (D)$ $(L) \leftrightarrow (E)$

Обмен содержимого регистровой пары H и указателя стека	XTHL	3	4	3	—	—	—	—	—	$(H, L) \leftrightarrow$ УС
Загрузка указателя стека	SPHL	3	7	1	—	—	—	—	—	$(H, L) \to$ УС
Загрузка счетчика команд	PCHL	3	5	1	—	—	—	—	—	$(H, L) \to$ СК
Двойное сложение содержимого регистровой пары H с парами B, D, H, SP	DADB	0	1	1	+	—	—	—	—	$(H, L) + (B, C) \to H,$
	DADD	0	3	1	+	—	—	—	—	$(H, L) + (D, E) \to H,$
	DADH	0	5	1	+	—	—	—	—	$(H, L) + (H, L) \to H,$
	DADSP	0	7	1	+	—	—	—	—	$(H, L) + (S, P) \to H,$
Инвертирование аккумулятора	CMA	0	5	7	—	—	—	—	—	$(\overline{A}) \to A$
Установка переноса	STC	0	6	7	1	—	—	—	—	$1 \to C$
Инвертирование переноса	CMC	0	7	7	+	—	—	—	—	$\overline{C} \to C$
Десятичная настройка аккумулятора	DAA	0	4	7	+	+	+	+	+	Если $\lvert A0 \ldots A3 \rvert > 9$ при $(C) = 1$, то $(A) + 6 \to A$; Если $\lvert A4 \ldots A7 \rvert > 9$ при $(C) = 1$, то $A7, A6, A5, A4 + 6 \to A7, A6, A5, A4$
Ввод в стек содержимого регистровых пар B, D и H	PUSHB	3	0	5	—	—	—	—	—	$(B) \to$ (УС) $- 1$; $(C) \to$ (УС) $- 2$
	PUSHH	3	4	5	—	—	—	—	—	$(H) \to$ (УС) $- 1$; $(L) \to$ (УС) $- 2$
	PUSHD	3	2	5	—	—	—	—	—	$(D) \to$ (УС) $- 1$; $(E) \to$ (УС) $- 2$
Вывод из стека двух слов в пары регистров B, D, H, PSW	POPB	3	0	1	—	—	—	—	—	(УС) $\to$ [(УС) $+ 1] \to$ } RP
	POPD	3	2	1						
	POPH	3	4	1						
Возврат из подпрограммы	RET	3	1	1	—	—	—	—	—	[(УС) $+ 1$], [УС] $\to$ СК (УС) $+ 2 \to$ УС
Возврат при условии:										
$(C) = 1$	RC	3	3	0	—	—	—	—	—	При выполнении условия [(УС) $+ 1$], [(УС)] $\to$ СК, иначе [(СК) $+ 1$] $\to$ СК
$(C) = 0$	RNC	3	2	0	—	—	—	—	—	
$(Z) = 1$	RZ	3	1	0	—	—	—	—	—	
$(Z) = 0$	PNZ	3	0	0	—	—	—	—	—	
$(S) = 0$	RP	3	6	0	—	—	—	—	—	
$(S) = 1$	RM	3	7	0	—	—	—	—	—	
$(P) = 1$	RPE	3	5	0	—	—	—	—	—	
$(P) = 0$	RPO	3	4	0	—	—	—	—	—	При выполнении условия [(УС) $+ 2$] $\to$ УС, иначе [(СК) $+ 1$] $\to$ СК
Повторный запуск	RST	3	i	7	—	—	—	—	—	(СК) $\to$ [(УС) $- 1$] [(УС) $- 2$], (УС) $- 2 \to$ УС, (00 i 0000) $\to$ СК
Разрешение прерывания	EI	3	7	3	—	—	—	—	—	ТгПр $= 1$

Продолжение табл. 3.9

Команда	Мнемоника	Код	Признаки					Содержание команды
			C	Z	S	P	C'	
Запрет прерывания	DI	3 6 3	—	—	—	—	—	ТгПр = 0
Останов	HLT	1 6 6	—	—	—	—	—	(СК) + 1 → СК
Холостая команда	NOP	0 0 0	—	—	—	—	—	—
Посылка в регистр второго байта команды	M ∨ IR ⟨B2⟩	0 R 6 ⟨B2⟩	— —	— —	— —	— —	— —	⟨B2⟩ → R —
Сложение A со вторым байтом	ADI ⟨B2⟩	3 0 6 ⟨B2⟩	+	+	+	+	+	(A) + ⟨B2⟩ → A
Сложение с переносом	ACI ⟨B2⟩	3 1 6 ⟨B2⟩	+	+	+	+	+	(A) + ⟨B2⟩ + C → A
Вычитание из A второго байта команды	SUI ⟨B2⟩	3 2 6 ⟨B2⟩	+	+	+	+	+	(A) — ⟨B2⟩ → A
Вычитание с заемом	SBI	3 3 6	+	+	+	+	+	(A) — ⟨B2⟩ — C → A
Логическое умножение	ANT ⟨B2⟩	3 4 6 ⟨B2⟩	0	+	+	+	0	(A) ∧ ⟨B2⟩ → A
Отрицание равнозначности	XRI ⟨B2⟩	3 5 6 ⟨B2⟩	0	+	+	+	0	(A) ⊕ ⟨B2⟩ → A
Логическое сложение	ORI ⟨B2⟩	3 6 6 ⟨B2⟩	0	+	+	+	0	(A) ∨ ⟨B2⟩ → A
Сравнение	CPI ⟨B2⟩	3 7 6 ⟨B2⟩	+	+	+	+	+	(A) — ⟨B2⟩
Ввод	IN ⟨B2⟩	3 3 3 ⟨B2⟩	—	—	—	—	—	⟨B2⟩ → A
Вывод	OUT ⟨B2⟩	3 2 3 ⟨B2⟩	—	—	—	—	—	(A) → ⟨B2⟩
Безусловный переход	JMP ⟨B2⟩ ⟨B3⟩	3 0 3 ⟨B2⟩ ⟨B3⟩	—	—	—	—	—	⟨B3⟩, ⟨B2⟩ → СК
Переход при условии:								

$(C)=1$	JC	3	3	2	—	—	—	—	—	При выполнении условия, иначе (СК) + 3 → СК
$(C)=0$	JNC	3	2	2	—	—	—	—	—	При выполнении условия $\langle B3\rangle\ \langle B2\rangle$ → СК, иначе (СК)+3 → СК
$(S)=0$	JP	3	6	2	—	—	—	—	—	
$(S)=1$	JM	3	7	2	—	—	—	—	—	
$(P)=1$	JPE	3	5	2	—	—	—	—	—	
$(P)=0$	JPO	3	4	2	—	—	—	—	—	
$(Z)=1$	JZ	3	1	2	—	—	—	—	—	
$(Z)=0$	JNZ	3	0	2	—	—	—	—	—	
Вызов подпрограммы безусловный	CALL	3	1	5	—	—	—	—	—	(СК) → [(УС) — 1]; [(УС) — 2], (УС) — 2 → УС
Вызов подпрограммы при условии:										
$(C)=1$	CC	3	3	4	—	—	—	—	—	При выполнении условия (см. команду CALL), иначе (СК) + 3 → СК
$(C)=0$	CNC	3	2	4	—	—	—	—	—	
$(Z)=1$	CZ	3	1	4	—	—	—	—	—	
$(Z)=0$	CNZ	3	0	4	—	—	—	—	—	
Загрузка памяти (прямая)	STA	0	6	2	—	—	—	—	—	$A \to [\langle B3\rangle\ \langle B2\rangle]$
Загрузка A (прямая)	LDA	0	7	2	—	—	—	—	—	$[\langle B3\rangle\ \langle B2\rangle] \to A$
Заполнить H и L по адресу $[\langle B3\rangle\ \langle B2\rangle]$	SHLD	0	4	2	—	—	—	—	—	$(L) \to [\langle B3\rangle\ \langle B2\rangle]$ $(H) \to [\langle B3\rangle\ \langle B2\rangle]$
Загрузить H и L содержимым ячейки памяти $[\langle B3\rangle\ \langle B2\rangle]$	LHLD	0	5	2	—	—	—	—	—	$(\langle B3\rangle\ \langle B2\rangle) \to L$ $(\langle B3\rangle\ \langle B2\rangle) \to H$
Сравнение	CMPR	2	7	R	+	+	+	+	+	Операнды (A) и (R): если $(A)=(R)$, то $(Z)=1$; если $(A)\neq(R)$, то $(C)=1$
Сдвиг влево с переносом	RAL	0	2	7	+	—	—	—	—	$2A \to A$, $(C) \to A0$, $A7 \to C$

Примечание. Здесь использованы следующие сокращения: (R) — содержимое регистра R; $\oplus$ — неравнозначность; → — переслать в ...; $\langle B_i\rangle$ — i-й байт команды; $[(R)]$ — адресом ячейки ЗУ является содержимое регистра R; ↔ — взаимный обмен содержимым регистров; SP (УС) — указатель стека (16-разрядный регистр) эквивалентен регистровой паре; СК — счетчик команд; $\langle R\rangle$ — двоичный код регистра; i и k — номера внутренних регистров; $R1$, $R2$ — внутренние регистры промежуточного хранения; ША, ШД — шины адреса и данных; ВС — вывод информации состояния.

п. 10. Действие команды приведено в табл. 3.9;
п. 11. *C* — разряд переноса;
п. 12. МЦ «Запись в стек»;
п. 13. МЦ «Чтение стека»;
п. 14. МЦ5 имеет пять тактов. Используются только в команде XTH;
п. 15. В МЦ1 следующей команды выполняются действия: (*W*, *Z*) → ША, ВС; (*W*, *Z*) + 1 → СК;
п. 16. Если условие не выполнено, то МЦ2 и МЦ3 пропускаются, и МП переходит к МЦ1 следующей команды.

В табл. 3.8 приведены статические параметры, а в табл. 3.9 — система команд МП.

3.3. МИКРОСХЕМА КР580ИК51

Микросхема представляет собой универсальный программируемый синхронно-асинхронный приемопередатчик (УСАП), выполняющий преобразование байта информации из параллельного кода в последовательный и наоборот.

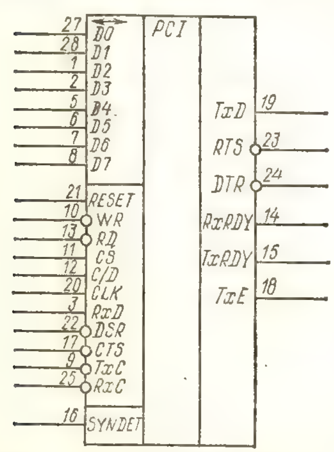


Рис. 3.14. Условное графическое обозначение микросхемы КР580ИК51

Микросхема приемопередатчика содержит 8-разрядную двунаправленную шину данных, двунаправленный последовательный канал приема-передачи данных, 11 входных и 7 выходных управляющих выводов. Приемопередатчик может работать как совместно с МП, так и под управлением другого активного устройства с синхронной и асинхронной дисциплинами обмена. Микросхема может работать в синхронном и асинхронном режимах. Кроме этого, по запросу с активного устройства приемопередатчик может выдавать слово-состояние, позволяющее определить наличие и вид ошибок обмена, а также ряд других ситуаций, требующих вмешательства процессора. На рис. 3.14 приведено условное графическое обозначение микросхемы. Назначение и обозначение выводов микросхемы приведены в табл. 3.10. Ниже даны латинские и отечественные обозначения выводов и сигналов микросхемы КР580ИК51: *RxD* — Вх Пр, *TxC* — СПр, *WR* — Зп, *CS* — ВУ, *C/D* — У/Д, *RD* — Чт, *RxRDY* — ГПр, *TxRDY* — ГПд, *SYNDET* — Вид Сн, *CTS* — ГПрТ, *TxE* — КПд, *TxD* — Вых Пд, *CLK* — Сн, *RESET* — *R*, *DSR* — ГПдТ, *PTS* — ЗПрТ, *DST* — ЗПдТ, *RxC* — СПр.

Шина данных *D*7 ... *D*0 (*ШД*) — двунаправленная трехстабильная, предназначена для передачи (приема) данных, управляющих слов и информации состояния; *D*0 — младший разряд шины.

Таблица 3.10. **Назначение выводов микросхемы КР580ИК51**

Номер вывода	Назначение	Разряд	Обозначение	Тип
1	Шина данных	2	$D2$	Вход-выход
2	Шина данных	3	$D3$	Вход-выход
3	Вход приемника	—	ВхПр	Вход
4	Общий	—	GND	—
5	Шина данных	4	$D4$	Вход-выход
6	Шина данных	5	$D5$	Вход-выход
7	Шина данных	6	$D6$	Вход-выход
8	Шина данных	7	$D7$	Вход-выход
9	Синхронизация передатчика	—	СПд	Вход
10	Запись	—	Зп	Вход
11	Выбор микросхемы	—	ВУ	Вход
12	Управляющие сигналы/данные	—	У/Д	Вход
13	Чтение	—	Чт	Вход
14	Готовность приемника	—	ГПр	Выход
15	Готовность передатчика	—	ГПд	Выход
16	Вид синхронизации	—	Вид Сн	Вход-выход
17	Готовность приемника терминала	—	ГПрТ	Вход
18	Конец передачи	—	КПд	Выход
19	Выход передатчика	—	Вых Пд	Выход
20	Синхронизация	—	Сн	Вход
21	Сброс	—	R	Вход
22	Готовность передатчика терминала	—	ГПдТ	Вход
23	Запрос приемника терминала	—	ЗПрТ	Выход
24	Запрос передатчика терминала	—	ЗПдТ	Выход
25	Синхронизация приемника	—	СПр	Вход
26	Питание	—	U_{cc}	—
27	Шина данных	0	$D0$	Вход-выход
28	Шина данных	1	$D1$	Вход-выход

Примечание. Шина данных трехстабильная

Входной сигнал *R* служит для установки УСАП в исходное состояние, которое будет сохраняться до прихода управляющего слова.

Входной сигнал *Сн* предназначен для запуска схемы внутренней синхронизации. Чаще всего сигналом *Сн* служит *Ф2* микропроцессорной системы.

Входной сигнал *Чт* разрешает передачу информации из УСАП по *ШД*. Наименование сигнала соотнесено с МП, поскольку из УСАП читает именно он. Это же относится и к входному сигналу *Зп*, который разрешает передачу информации из МП в УСАП.

Входной сигнал *У/Д* устанавливает вид передаваемой или принимаемой УСАП информации.

Входной сигнал *ВУ* разрешает работу данной микросхемы.

Выходной сигнал *Вых Пд* является очередным битом информации, передаваемой из УСАП в последовательном коде. Прием последовательной информации осуществляется по входу *Вх Пр*.

Входной сигнал *СПд* стробирует (отрицательным фронтом) биты информации, поступающие с *Вых Пд*.

Выходной сигнал *ЗПдТ* может быть использован как запрос о готовности внешнего устройства передавать данные.

Выходной сигнал *ЗПрТ* может являться запросом готовности внешнего устройства принять данные. Выходной сигнал *КПд*

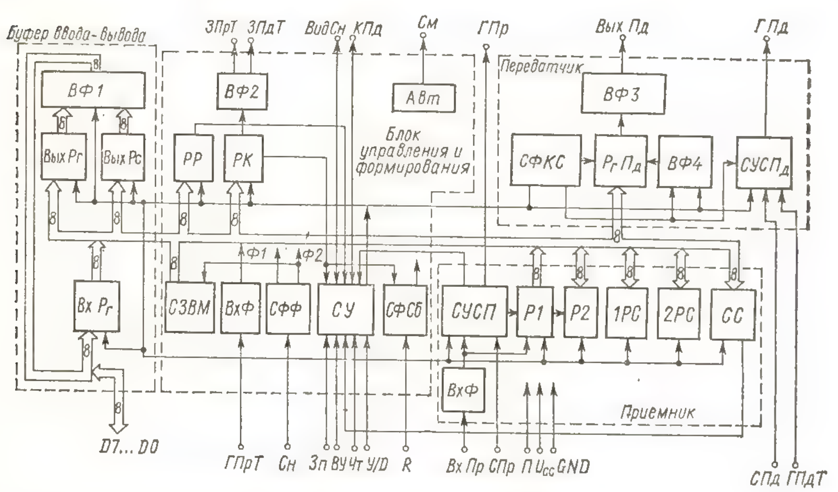


Рис. 3.15. Структурная схема микросхемы КР580ИК51

фиксирует окончание посылки данных. Он автоматически сбрасывается после получения данных из МП. Выходной сигнал *ГПр* свидетельствует о готовности приемника УСАП передать данные на *ШД*. Выходной сигнал *ГПд* свидетельствует о готовности УСАП принять данные из МП по *ШД*.

Входной сигнал *ГПрТ* свидетельствует о готовности внешнего устройства принять данные.

Сигнал является ответом на сигнал *ЗПрТ*. Входной сигнал *ГПдТ* свидетельствует о готовности внешнего устройства передать данные. Сигнал является ответом на сигнал *ЗПдТ*.

Сигнал *Вид Сн* может быть как входным, так и выходным. Направление передачи определяется программно.

В состав структурной схемы УСАП входят (рис. 3.15): приемник, передатчик, блок управления и формирования, буфер ввода-вывода.

Буфер ввода-вывода представляет собой трехстабильное 8-разрядное устройство, предназначенное для организации двунаправленной связи (параллельным кодом) УСАП и МП.

Передатчик состоит из схем выходного формирователя *ВФ3*, *ВФ4*, схемы управления передатчиком *СУСПд*, регистра передатчика *РгПд*, схемы формирования конца слова *СФКС*.

Приемник состоит из входного формирователя, регистров *Р1* и *Р2*, регистров первого и второго слова *1РС* и *2РС*, схемы управления *СУСП*, схемы синхронизации *СС*.

Буфер содержит входной регистр *ВхРг*, выходной регистр данных *Вых Рг*, выходной регистр состояния *Вых РС*, выходные формирователи *ВФ1*.

Блок управления и формирования предназначен для выдачи внутренних и ряда внешних управляющих сигналов. Он состоит из выходного формирователя *ВФ2*, регистров режима и команд *РР* и *РК*, а также схем управления *СУ*, формирования фаз *Ф1* и *Ф2* *СФФ*, формирования сброса *СФС*, автосмещения *Авт* и зарядки внутренней магистрали *СЗВМ*.

РР и *РК* — 8-разрядные регистры, предназначенные для хранения инструкции режима и инструкции команды соответственно.

ВФ2 позволяет посылать запросы на обмен с внешними устройствами (2-й и 6-й разряды *РК*). *СУ* вырабатывает сигналы управления для всех основных схем УСАП. *СФСб* устанавливает УСАП в исходное состояние по сигналу *R* либо программно. Схема *Авт* предназначена для выработки внутреннего сигнала смещения. Схема состоит из *RC*-генератора, формирователя и фильтра. *СЗВМ* (как и для МП) предназначена для повышения быстродействия микросхемы. Она содержит 8 транзисторов, затворы которых управляются импульсами *Ф2*. Транзисторы коммутируют напряжение питания на линии внутренней магистрали УСАП, ускоряя тем самым процессы ее заряда и разряда.

Подготовка УСАП к работе осуществляется в определенном порядке. Подается сигнал *R*, который переводит УСАП в исходное состояние. Затем осуществляется программирование УСАП. На первом этапе программирования по *ШД* в УСАП заносится инструкция режима. Занесение происходит при наличии фазовых импульсов *Ф2* на входе *Сн* и при условии, что сигналы *У/Д*, *Зп*, *ВУ* — в состоянии «0», сигнал *Чт* — в состоянии «1»; сигнал *Зп* — в состоянии «0». Формат инструкции режима приведен на рис. 3.16. Эта инструкция определяет и режим работы УСАП в данном цикле передачи. Режимы могут быть синхронными или асинхронными. При реализации синхронного режима за инструкцией следуют служебные слова — идентификаторы информации (синхросимволы). Число синхросимволов определено двумя старшими разрядами инструкции режима. Кроме этого, инструкция режима задает вид синхронизации, определяет вид контроля передачи, а также длину передаваемого слова. Направление передачи задается внешними сигналами *Чт* и *Зп*. Первый из них настраивает УСАП на передачу информации из УСАП на *ШД*. Второй — задает обратное направление.

Асинхронный режим			
Длина стоп-бита (бит)			
2	1,5	1	сбой
1	0	1	0
1	1	0	0

Асинхронный режим		
1:1	1:16	1:64
1	0	1
0	1	1

D7 D6 D5 D4 D3 D2 D1 D0

Синхронный режим
Число синхросимволов
1 – один 0 – два
Вид синхронизации
1 – внешняя 0 – внутренняя

Синхронный режим
0
0

Длина слова (бит)			
5	6	7	8
0	1	0	1
0	0	1	1

Контроль передач
1 – четность 0 – нечетность
1 – есть 0 – нет

Рис. 3. 16. Формат инструкции режима

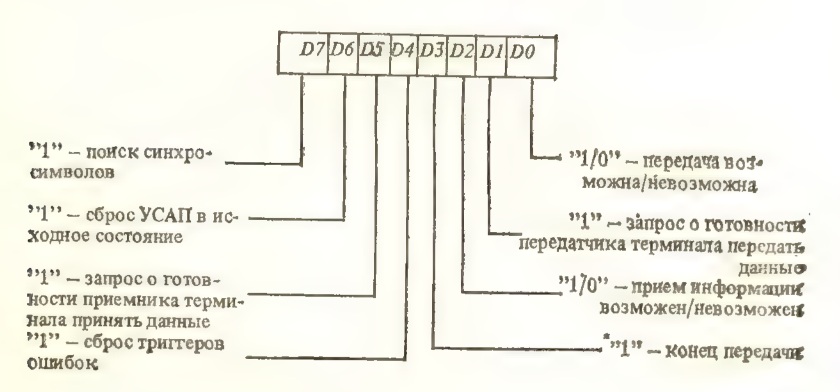


Рис. 3.17. Формат инструкции команды

Вид информации (управляющие слова или данные) определяется сигналом *У/Д*. Переводом ШД УСАП в третье состояние управляет сигнал ВУ.

Итак, вслед за синхросимволом (синхросимволами) в синхронном режиме работы следует инструкция команды, поступающая по *ШД* в УСАП. Она программно определяет возможность приема и передачи информации, выдачу запросов о готовности провести обмен, а также другие действия УСАП (рис. 3.17). После записи инструкции команды УСАП выдает сигнал *ГПд*-«1», синхронизируя о готовности микросхемы принять данные из МП. Сброс этого сигнала происходит после подачи сигнала *Зп*-«0» и данных по *ШД*. Инструкциями режима и команды УСАП переводятся в один из пяти режимов работы: асинхронная передача, асинхронный прием, синхронная передача, синхронный прием, чтение состояния.

В режиме «Асинхронная передача» принятые УСАП по *ШД* данные выдаются в последовательном коде на вывод *Вых Пд*. Перед посылкой выводится старт-бит. После передачи байта данных может быть передан бит контроля, а затем — стоп-бит. Наличие и вид контроля, а также длительность стоп-бита определяются инструкцией режима. Соотношение между скоростью появления информации на *Вых Пд* и частотой появления сигналов *Спд* определяется инструкцией режима (см. рис. 3.16, 3.18).

В режиме «Асинхронный прием» напряжение «0» на входе *ВхПр* (рис. 3.19) свидетельствует о приходе старт-бита последовательной посылки. Истинность этого бита проверяется вторично стробированием в его середине (рис. 3.20). После этого запускается внутренний счетчик битов, позволяющий определить конец посылки. Если при вторичной проверке обнаружится напряжение «1», то УСАП переходит в исходное состояние. При обнаружении ошибок передачи выводится состояние соответствующего внутреннего триггера. Состояние внутренних триггеров УСАП программно доступно МП.

Ошибки не останавливают работу УСАП.

Принятые данные передаются в выходной регистр данных, и появляется сигнал *ГПр*-«1».

В режиме «Синхронная передача» перед битами данных вводятся синхросимволы, предназначенные для выделения собственно данных из потока информации. Как было отмечено, синхросимволы следуют за инструкцией режима. После записи инструкции режима, синхросимволов, инструкции команд и данных происходит анализ сигнала *ГПрТ* и разряда *D*0 инструкции команды. При наличии разрешающих сигналов УСАП транслирует принятую информацию на вывод *ВхПд* синхронно с сигналами *СПд*. В случае, если в УСАП не поступят очередные данные до того, как он передаст предыдущую информацию, в поток данных автоматически будут включены синхросимволы. При этом появится сигнал *КПд*, идентифицирующий передачу синхросигналов.

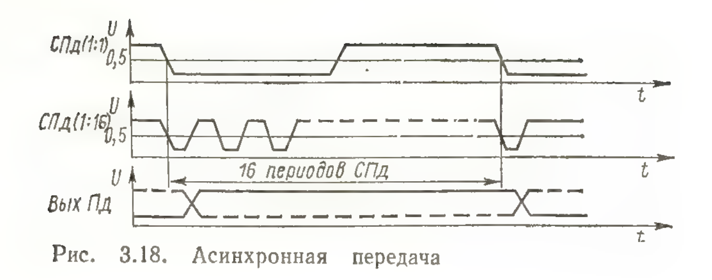


Рис. 3.18. Асинхронная передача

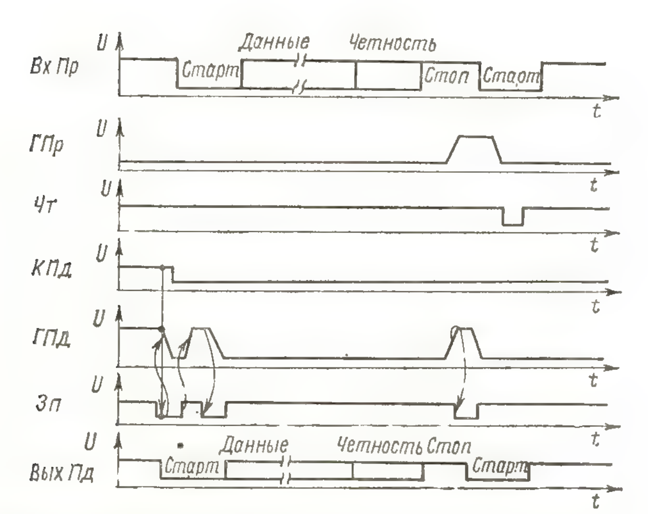


Рис. 3.19. Формирование сигналов *ГПд* и *ГПр* в асинхронных режимах работы УСАП

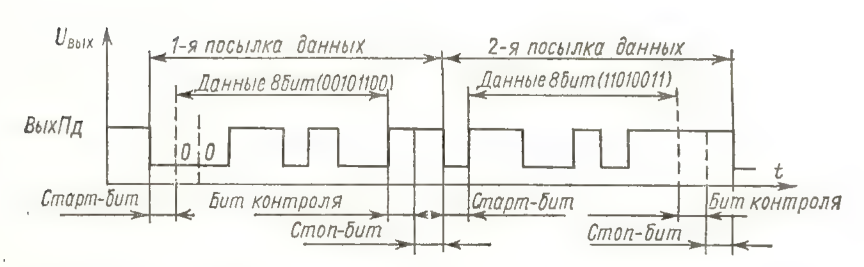


Рис. 3.20. Формат передачи (приема) данных в асинхронном режиме

Режим «Синхронный прием» может быть реализован как с внутренней, так и с внешней синхронизацией (рис. 3.21 и 3.22).

С внутренней синхронизацией информация принимается по входу *ВхПр* по положительному фронту сигнала СПр. Вначале происходит поиск синхросимволов. На этом этапе информация,

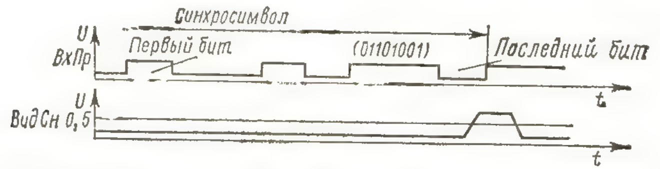


Рис. 3.21. Синхронный прием данных с внутренней синхронизацией

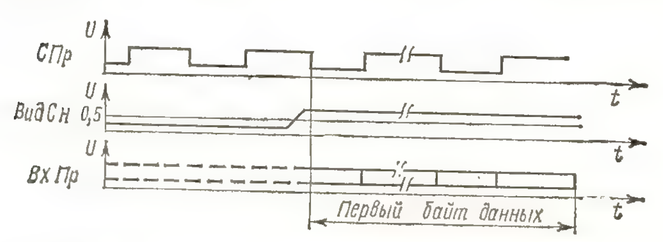


Рис. 3.22. Синхронный прием данных с внешней синхронизацией

поступившая на первый (второй) регистр приемника, сравнивается с содержимым регистра первого (второго) синхросимвола. После этого (одновременно с приемом последнего бита послед-

Таблица 3.11. **Статические параметры микросхемы КР580ИК51**

Параметр, единица измерения	Обозначение	Значения			Режим измерения
		мин.	ном.	макс.	
Входное напряжение «1», В	U_{IH}	2,0	3,0	5,5	—
Входное напряжение «0», В	U_{IL}	—0,5	0,5	0,8	—
Выходное напряжение «1», В	U_{OH}	2,4	—	—	I_{OH} = —0,4 мА
Выходное напряжение «0», В	U_{OL}	—	0,4	0,45	I_{OL} = 8,2 мА
Ток потребления, мА	I_{CC}	—	80	100	U_{CC} = +5,25 В
Напряжение питания, В	U_{CC}	4,75	5,0	5,25	—
Ток утечки по входам, мкА	I_{IZ}	—	—	±10	$0 \leqslant U_{OZ} \leqslant U_{CC}$
Ток утечки по входам-выходам, мкА	I_{IOZ}	—	—	±10	То же

Примечание. Параметры приведены при напряжении питания U_{CC} = 5 В (±5 %) и температуре окружающей среды 25 °С

него синхросимвола) выдается сигнал *Вид Сн* к терминалу. Сброс этого сигнала происходит в режиме «Чтение состояния».

В режиме «Синхронный прием с внешней синхронизацией» сигнал Вид Сн является входным (в отличие от указанного режима с внутренней синхронизацией), он синхронизирует подачу синхросимвола (синхросимволов) в УСАП. При условии *Вид Сн*=«0» прием информации задерживается. Следует отметить, что ошибки четности и переполнения контролируются так же, как и в асинхронных режимах.

Режим «Чтение состояния» используется для определения состояния УСАП в процессе выполнения операций. В этом режиме можно выдать из УСАП на *ШД* содержимое регистра состояния УСАП.

В табл. 3.11 приведены статические параметры УСАП.

3.4. МИКРОСХЕМА КР580ВИ53

Микросхема представляет собой однокристальный трехстабильный программируемый таймер (ПТ), предназначенный для получения программно-управляемых временных задержек и выполнения времязадающих функций в микропроцессорных системах.

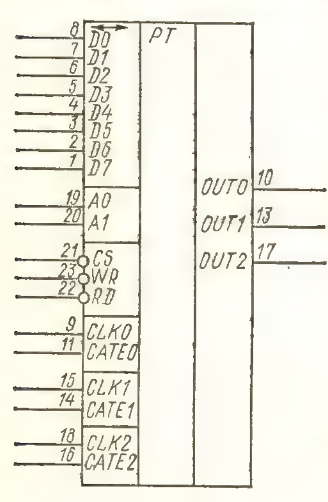


Рис. 3.23. Условное графическое обозначение микросхемы КР580ВИ53

Микросхема применяется в системах обработки информации, выполненных на основе микропроцессора КР580ИК80А, и позволяет повысить эффективность программирования процессов управления и синхронизации внешних устройств, особенно в реальном масштабе времени. Кроме того, микросхема может применяться как самостоятельное устройство при условии выполнения требований, предъявляемых к ее электрическим и временным параметрам.

Условное графическое обозначение микросхемы представлено на рис. 3.23, а назначение выводов — в табл. 3.12.

Ниже даны латинские и отечественные обозначения выводов и сигналов микросхемы КР580ВИ53: *CS* — ВМ, *WR* — *Зп*, *RD* — Чт, *OUT* — Вых, *CLK* — ТИ, *CATE* — Р.

Микросхема имеет три независимых канала.

Шина данных *ШД* (*D*7 ... *D*0) — двунаправленная трехстабильная, предназначенная для приема управляющих слов и данных из микропроцессора и передачи данных в виде показаний счетчиков из микросхемы в микропроцессор, *D*0 — младший разряд.

Таблица 3.12. **Наименование и назначение выводов микросхемы КР580ВИ53**

Номер вывода	Назначение	Разряд	Обозначение	Тип
1	Шина данных	7	$D7$	Вход-выход
2		6	$D6$	
3		5	$D5$	
4		4	$D4$	
5		3	$D3$	
6		2	$D2$	
7		1	$D1$	
8		0	$D0$	
9	Тактовый импульс канала 0	—	ТИ0	Вход
10	Выход канала 0	—	Вых 0	Выход
11	Сигнал разрешения или запрещения работы канала 0	—	Р0	Вход
12	Общий	—	GND	—
13	Выход канала 1	—	Вых 1	Выход
14	Сигнал разрешения или запрещения работы канала 1	—	Р1	Вход
15	Тактовый импульс канала 1	—	ТИ1	
16	Сигнал разрешения или запрещения работы канала 2	—	Р2	
17	Выход канала 2	—	Вых 2	Выход
18	Тактовый импульс канала 2	—	ТИ2	Вход
19	Адрес	0	А0	
20	Адрес	1	А1	Вход
21	Выбор микросхемы	—	ВМ	
22	Сигнал разрешения чтения	—	Чт	
23	Сигнал разрешения записи	—	Зп	
24	Питание +5В	—	U_{cc}	—

Примечание. Шина данных трехстабильная.

ТИ2 ... ТИ0 — входные сигналы, предназначенные для синхронной работы счетчиков каналов 2, 1, 0 соответственно.

Вых2 ... Вых0 — выходные сигналы соответственно 0, 1, 2 каналов, *Р2 ... Р0* — входные сигналы соответствующих каналов (2, 1, 0) разрешают или запрещают работу счетчика.

ВМ — «0» на этом входе разрешает работу данной микросхемы.

Чт — сигнал «0» на этом входе разрешает передачу информации из микросхемы в шину данных МП в виде показаний счетчиков.

Зп — сигнал «0» на этом входе разрешает микросхеме принимать информацию из МП в виде управляющих слов или данных.

Микросхема состоит из следующих функциональных узлов (рис. 3.24): буфера шины данных *Буф ШД*, схемы выбора канала *СВК*, схемы управления чтением «на лету» *СУЧЛ*, схем каналов 0, 1, 2 (последние схемы на рис. 3.24 не обозначены).

Буф ШД представляет собой трехстабильную двунаправленную схему, предназначенную для сопряжения ПТ с шиной данных МП.

СВК предназначена для формирования сигналов управления каналами 0, 1, 2, внутренними и внешними передачами данных и управляющих слов, а также для выполнения других функций управления. Если данная микросхема не выбрана, то операции записи и чтения невозможны. *СУЧЛ* позволяет прочитать содер-

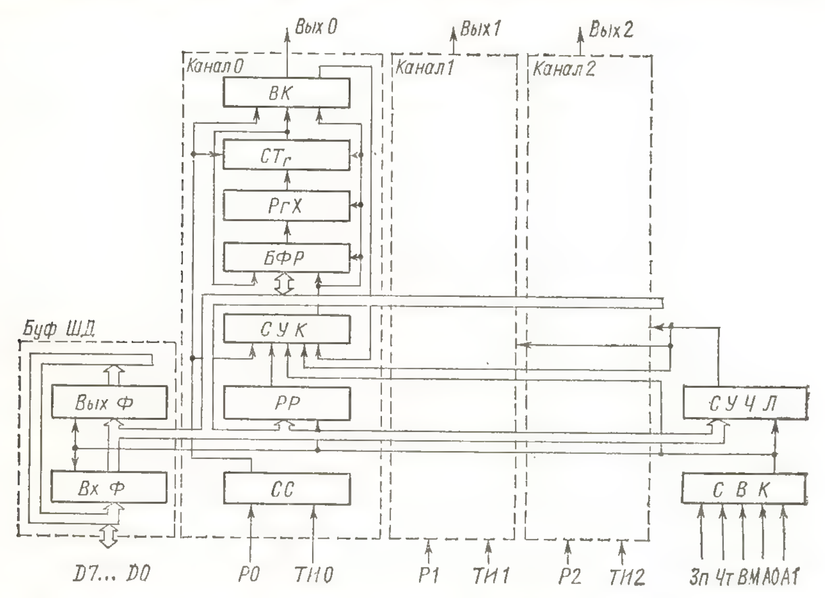


Рис. 3.24. Структурная схема микросхемы КР580ВИ53

жимое счетчиков, не прерывая текущего счета. Схемы каналов идентичны. Каждый канал содержит схему синхронизации *СС*, регистр режима *РР*, схему управления *СУК*, выходной каскад *ВК*, счетный триггер *СТг*, буферный регистр *БФР* и регистр хранения *РгХ*. Последние три схемы образуют счетчик.

Схема *СС* формирует серию внутренних тактовых импульсов определенной длительности. Длительность этих импульсов не зависит от внешней частоты синхронизации и определяется внутренними времязадающими цепями, а период равен периоду внешней частоты синхронизации.

Регистр *РР* (6-разрядный регистр) предназначен для приема и хранения кода управляющего слова, который задает режим работы канала, определяет тип счета (двоичный или двоично-десятичный) и последовательность загрузки данных в счетчик.

Информацию в регистр режима можно только записывать. Прочитать содержимое невозможно.

Схема *СУК* синхронизирует работу отдельных схем в соответствии с запрограммированным режимом работы и работу канала с работой МП.

Счетчик представляет собой 16-разрядную вычитающую схему. Он предназначен для выполнения счетных операций в соответствии с запрограммированным режимом.

Счетчик выполняет счетные операции в десятичном или двоично-десятичном коде над однобайтовыми или двухбайтовыми числами.

Возможная частота счета от 0 до 2 МГц. Максимальная величина счета в двоичном коде — 2^{16}, в двоично-десятичном — 10^4. Счетчики трех каналов независимы друг от друга и могут иметь различные режимы работы и режимы счета (двоичный или двоично-десятичный).

БФР принимает данные, поступающие из МП, и передает их в *РгХ* или принимает информацию из *СТг* и передает ее МП.

В *РгХ* информация поступает из *БФР* (обратная операция невозможна). Содержимое *РгХ* изменяется только при перезагрузке счетчика или при перепрограммировании режима работы канала (запись управляющего слова в регистр режима). В последнем случае регистр хранения устанавливается в нулевое состояние.

СТг принимает информацию из *РгХ*. Поскольку счетчик вычитающий, число, загруженное в *СТг*, будет декрементироваться.

Загрузка во все разряды *СТг* информации в виде нулей позволяет получить максимально возможную величину света. Информация из *СТг* передается в *БФР* и при необходимости может быть прочитана.

ВК выполняет функции времязадающего элемента и обеспечивает согласование выхода данной микросхемы с одним входом микросхемы ТТЛ-типа.

ПТ программируется МП системы. Для приведения каждого канала ПТ в исходное состояние (в соответствии с выбранным режимом) и для загрузки ПТ информацией о величине счета МП должен послать в ПТ набор управляющих слов, которые программируют режим, очередность загрузки и тип счета. После программирования ПТ готов к выполнению задач, связанных с отсчетом времени.

Режим работы ПТ программируется с помощью простых операций ввода-вывода (табл. 3.13). Каждый из трех каналов ПТ программируется отдельно, путем записи в регистр режима управляющего слова и запрограммированного количества байтов.

Разряд *D*0 управляющего слова определяет двоичный или двоично-десятичный тип счета. Разряды *D*1, *D*2, *D*3 задают двоичным кодом режим работы канала. Разряды *D*4, *D*5 определяют количество байтов информации, необходимой для загрузки счетчика или чтения его показаний. Сочетание сигналов *D*4 = «0», *D*5 = «0» — особая команда, позволяющая при чтении «на лету»

Таблица 3.13. **Действия производимые входными сигналами микросхемы КР580ВИ53**

Входные сигналы					Направление и вид информации
Зп	Чт	A1	A0	ВМ	
0	1	1	1	0	ШД → ПТ. Занесение управляющего слова в каналы 0, 1, 2
1	0	1	1	0	Нет операций. Канал данных ПТ в третьем состоянии
0	1	0	0	0	ШД → ПТ. Загрузка счетчика канала 0
0	1	0	1	0	ШД → ПТ. Загрузка счетчика канала 1
0	1	1	0	0	ШД → ПТ. Загрузка счетчика канала 2
1	0	0	0	0	ПТ → ШД. Чтение показаний счетчика канала 0
1	0	0	1	0	ПТ → ШД. Чтение показаний счетчика канала 1
1	0	1	0	0	ПТ → ШД. Чтение показаний счетчика канала 2
1	1	X	X	0	Нет операций. Шина данных ПТ в третьем состоянии
X *	X	X	X	1	Запрет. Шина данных ПТ в третьем состоянии.

* X — состояние входа безразлично.

прочитать содержимое счетчика, не прерывая процесса счета (выполнить операцию «Защелкивание»). Разряды *D*6, *D*7 служат адресом для выбора регистра режима соответствующего канала при записи управляющего слова или для выбора канала при вводе команды чтения «на лету».

Адресация каналов приведена в табл. 3.14.

Режим работы каждого канала ПТ определяется содержимым регистра режима. При записи управляющего слова в регистр режима выбранного канала последний переводится в один из шести основных режимов работы: 0 — прерывание терминального счета;

Таблица 3.14. **Адресация каналов микросхемы КР580ВИ53**

Операция	Канал	Адресация	
		A1	A0
Запись управляющего слова	0	1	1
Запись управляющего слова	1	1	1
Запись управляющего слова	2	1	1
Загрузка младшего байта в счетчик	1	0	1
» старшего » » »	1	0	1
» младшего » » »	2	1	0
» старшего » » »	2	1	0
» младшего » » »	0	0	0

1 — ждущий мультивибратор; 2 — генератор частоты импульсов; 3 — генератор меандра; 4 — одиночный программно-формируемый строб; 5 — одиночный аппаратно-формируемый строб.

Временна́я диаграмма работы канала ПТ в режиме 0 (прерывание терминального счета) показана на рис. 3.25. В режиме 0 по окончании отсчета числа, загружаемого в счетчик, на выходе канала ПТ устанавливается «1» и сохраняется до загрузки счетчика новой счетной величиной.

Последовательность работы ПТ в режиме 0 следующая: после выполнения операции установки режима выбранного канала

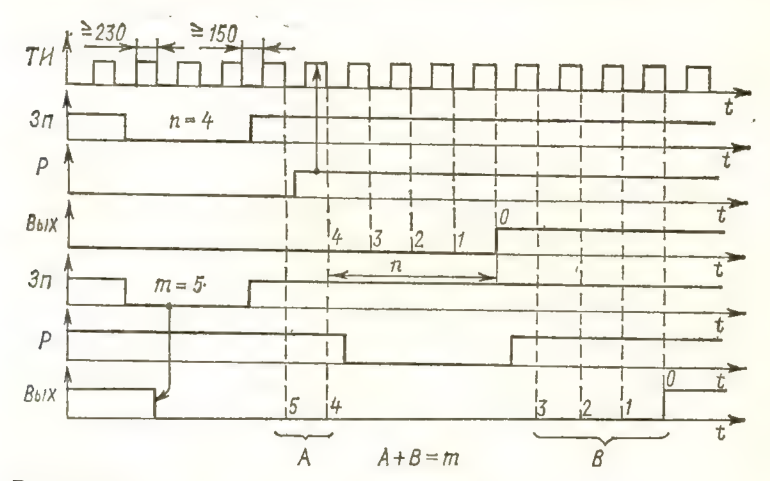


Рис. 3.25. Работа микросхемы в режиме 0

на *Вых* этого канала устанавливается «0»; загрузка счетчика не изменяет состояния на выходе. При подаче на вход *P* канала «1» счетчик начинает счет. По окончании счета на *Вых* устанавливается «1».

Загрузка счетчика новой счетной величиной изменяет «1» на «0». Перезагрузка счетчика во время счета приводит к тому, что загрузка младшего байта останавливает текущий счет или загрузка старшего байта запускает новый цикл счета.

Правильность загрузки счета можно проконтролировать, выполнив затем операцию чтения. Если во время загрузки счетчика новой счетной величиной отсутствуют сигналы на входе ТИ, то прочитать загруженное число нельзя, так как счетная величина будет находиться в регистре хранения и для ее передачи в счетный триггер необходим хотя бы один тактовый импульс.

В режиме 1 (рис. 3.26) канал ПТ формирует отрицательные импульсы длительностью $t_и = T_{ТИ}n$, где $T_{ТИ}$ — период тактовых импульсов; n — число, загруженное в счетчик.

На выходе вслед за положительным фронтом сигнала, подаваемого на вход P, устанавливается «0», который изменяется на «1» только после завершения счета. Если во время счета в счетчик будет загружена новая счетная величина, то она не повлияет на длительность текущего импульса ждущего мультивибратора до следующего запуска. Ждущий мультивибратор в данном случае является перезапускаемым, т. е. каждый положительный фронт сигнала P запускает счет или перезапускает его сначала, если не завершен счет до конца.

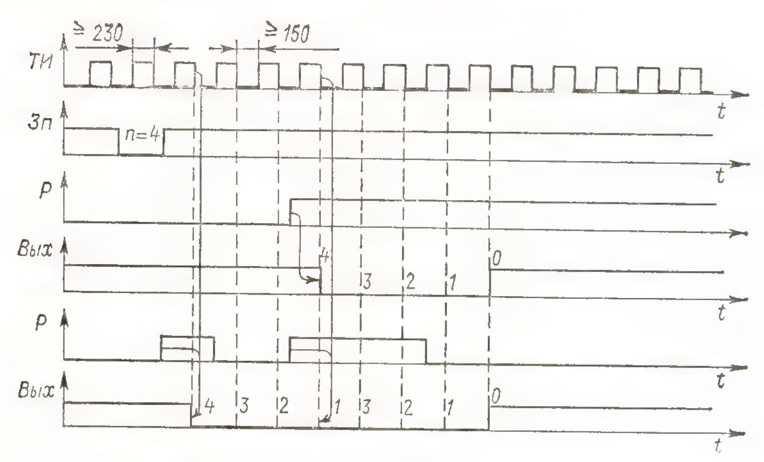


Рис. 3.26. Работа микросхемы в режиме 1

В режиме 1 содержимое регистра хранения передается в счетный триггер только при наличии сигнала ТИ и положительного фронта сигнала, подаваемого на вход P.

В режиме 2 (рис. 3.27) канал ПТ работает как делитель входной частоты $f_{ТИ}$ на число n. На выходе ПТ с частотой $f_{ТИ}/n$ устанавливается «0» на время одного периода сигнала ТИ. Если счетчик перезагружается между выходными импульсами, то на текущем периоде это не сказывается, однако последующий период будет соответствовать уже новой счетной величине.

При подаче на вход P «1» на выходе ПТ устанавливается «1», счетчик начинает отсчет от начальной величины счета, поэтому вход P можно использовать для синхронизации работы канала с внешними событиями.

Режим 3 аналогичен режиму 2 за исключением того, что длительность положительных и отрицательных полупериодов выходного сигнала для четных чисел равна

$$\tau = T_{ТИ}n/2, \qquad (3.1)$$

для нечетных положительных чисел — $T_{ТИ}n/2$, а для отрицательных — $T_{ТИ}(n-1)/2$. Перегрузка счетчика во время счета

новой счетной величиной не влияет на текущий период, но следующий период будет уже новым.

В режиме 4 по окончании отсчета числа, загруженного в счетчик, на выходе канала ПТ на время одного периода сигнала ТИ устанавливается «0», а затем снова «1».

При сигнале «1» на входе P разрешается счет, а при «0» — запрещается; т. е. по действию сигнала P режим 4 аналогичен режиму 0.

Перегрузка счетчика во время счета приводит к тому, что загрузка младшего байта не изменяет счетчик, загрузка старшего байта запускает новый цикл счета.

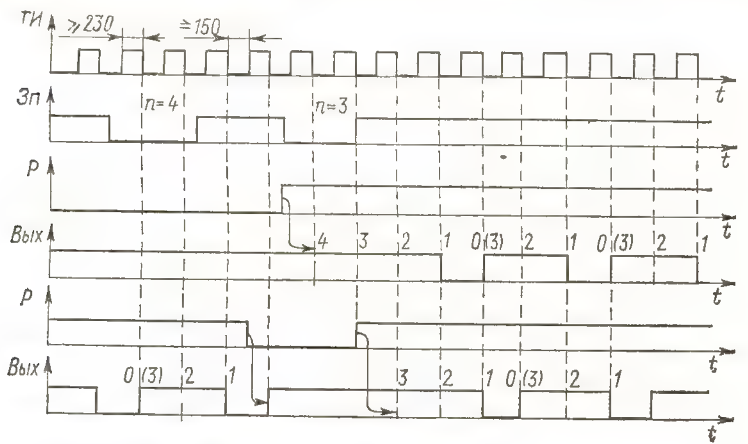


Рис. 3.27. Работа микросхемы в режиме 2

По выходному сигналу режим 5 аналогичен режиму 4, а по действию сигнала P — режиму 1.

Запуск счета осуществляется положительным фронтом сигнала P. Счетчик в этом режиме является перезапускаемым, т. е. каждый положительный фронт сигнала запускает счет или перезапускает его сначала, если счет не завершен до конца. Перезагрузка счетчика новой счетной величиной во время счета не влияет на длительность текущего цикла, но следующий цикл будет уже новым.

В зависимости от режима работы входной сигнал P выполняет функции, приведенные в табл. 3.15.

Режим работы и величина счета выбранного канала ПТ программируются путем записи в регистр режима управляющего слова, а в счетчик — необходимого количества байтов (одного или двух).

Ввод информации в ПТ осуществляется при наличии «0» на входе *Зп*.

Порядок программирования ПТ весьма гибок. Запись в ПТ управляющих слов возможна при наличии на входах $A0$ = «1»,

Таблица 3.15. Функции сигнала «Разрешение»

<table>
<tr><th>Режим</th><th>Отрицательный фронт или «0»</th><th>Положительный фронт</th><th>«1»</th></tr>
<tr><td>0</td><td>Запрещает счет</td><td>—</td><td>Разрешает счет</td></tr>
<tr><td>1</td><td>—</td><td>1. Запускает счет сначала
2. На Вых установлен «0» со следующего такта синхронизации сигнала ТИ</td><td>—</td></tr>
<tr><td>2</td><td>1. Запрещает счет
2. Немедленно устанавливает на выходе сигнала «1»</td><td rowspan="2">Запускает счет сначала</td><td rowspan="3">Разрешает счет</td></tr>
<tr><td>3</td><td>1. Запрещает счет
2. Немедленно устанавливает на выходе сигнал «1»</td></tr>
<tr><td>4</td><td>Запрещает счет</td><td>—</td></tr>
<tr><td>5</td><td>—</td><td>Запускает счет сначала</td><td>—</td></tr>
</table>

$A1$ = «1» и может производиться в каналы в любой последовательности, так как регистр режима в каждом канале имеет свой адрес, определяемый комбинацией значений разрядов $D6$, $D7$ в управляющем слове. Очередность загрузки счетчиков, так же как и при записи управляющих слов в регистр режима может быть произвольной (например, можно сначала запрограммировать режим работы каналов, а затем загружать счетчики). Однако если выбранный счетчик канала подлежит загрузке, то он обязательно должен быть загружен полностью именно тем количеством байтов, которое было запрограммировано в управляющем слове (комбинация значений $D5 \neq$ «0», $D4 \neq$ «0»).

В большинстве случаев использование счетчиков требует считывания тех или иных значений. Особенно это характерно для счета событий.

Во время работы счетчика величина текущего счета передается в буферный регистр и может быть прочитана двумя способами: выполнением обычной операции чтения или вводом специальной команды и последующим чтением (чтение «на лету»).

При первом способе для обеспечения стабилизации показаний счета работа счетчика должна быть приостановлена подачей на управляющий вход *P* «0» (режим 0, 2, 3, 4) или с помощью внешней логической схемы, которая прекратит подачу на счетчик сигналов ТИ.

Из-за наличия в ПТ специальной внутренней логической схемы операцию чтения содержимого счетчика необходимо обязательно выполнить до конца, т. е. если запрограммировано чтение двух байтов, то нельзя, прочитав один младший байт, перегружать счетчик новой счетной величиной. Содержимое счетчика в этом случае извлекают начиная с младшего байта. Следует помнить, что операция чтения невозможна при $A0 = A1 =$ «1».

Второй способ чтения заключается в том, что программист может считывать содержимое счетчика, не прерывая процесса счета. Для этой цели ПТ имеет специальную внутреннюю логическую схему, обращение к которой возможно с помощью записи определенного управляющего слова. Формат слова показан в табл. 3.16.

В разрядах *D*7, *D*6 указан номер канала, в котором нужно произвести операцию «Защелкивание». Разряды $D5 = 0$, $D4 = 0$ указывают на необходимость произвести операцию «Защелкивание»; X — состояние разрядов безразлично.

Адресные разряды *A0*, *A1* должны быть установлены в «1». Запись в ПТ управляющего слова для «Защелкивания» счета не влияет на содержимое регистра режима. По этой команде в буферном регистре защелкивается текущая величина счета, т. е. происходит ее запоминание в период ввода данной команды. Далее следует обычная операция. Для этого способа справедливо то же ограничение, что и для первого. Операция чтения обязательно должна быть завершена полностью в соответствии с запрограммированным числом байтов. После полного завершения операции чтения снимается запрет, и в буферный регистр поступает текущее значение счета.

При втором способе чтения нельзя предварительно отдельно по каждому каналу производить запись управляющих слов для «защелкивания» счета, а затем производить чтение.

Таблица 3.16. Формат управляющего слова для «чтения на лету» микросхемы КР580ВИ53

Канал	Разряды шины данных							
	*D*7	*D*6	*D*5	*D*4	*D*3	*D*2	*D*1	*D*0
0 1 2	0 0 1	0 1 0	0	0	X	X	X	X

Таблица 3.17. Статические параметры микросхемы КР580ВИ53

Параметр, единица измерения	Обозначение	Значения	
		мин.	макс.
Напряжение источника питания, В	U_{CC}	4,75	5,25
Ток потребления, мА	I_{CC}	—	140
Входное напряжение «1», В	U_{IH}	2,4	—
Входное напряжение «0», В	U_{IL}	—	0,4
Выходное напряжение «1», В	U_{OH}	2,4	—
Выходное напряжение «0», В	U_{OL}	—	0,45
Выходной ток «1», мА	I_{OH}	—	0,1
Выходной ток «0», мА	I_{OL}	—1,6	—
Ток утечки на входах, мкА	I_{IZ}	—	25
Ток утечки на входах-выходах, мкА	I_{IOZ}	—	50
Входная емкость, пФ	C_I	—	10
Емкость входа-выхода, пФ	C_{IO}	—	20
Емкость нагрузки, пФ	C_L	—	100

Примечание. Емкость нагрузки измеряется на частоте 1 МГц относительно вывода «Общий».

Содержимое счетчика при этом способе чтения выводят в следующем порядке:

операция записи «защелкивает» текущее значение счетчика в буферный регистр;

в первой операции чтения извлекается содержимое младшего байта;

во второй операции чтения извлекается содержимое старшего байта.

Статические параметры микросхемы приведены в табл. 3.17.

3.5. МИКРОСХЕМА КР580ИК55

Микросхема КР580ИК55 представляет собой БИС программируемого параллельного интерфейса (БИС ППИ) и предназначена для организации параллельного обмена между центральным обрабатывающим устройством (например, КР580ИК80А) и периферийным оборудованием. Обмен может происходить как байтами, так и полубайтами информации. Микросхема используется в качестве элемента ввода-вывода в блоках сопряжения с дисплеем, телетайпом, накопителем на магнитной ленте и т. д.

Условное графическое обозначение и структурная схема БИС ППИ приведены на рис. 3.28 и 3.29.

В табл. 3.18 приведено назначение выводов БИС ППИ.

Ниже даны латинские и отечественные обозначения выводов и сигналов микросхемы КР580ИК55: *PA-КА*, *PB-КВ*, *PC-КС*, *RESET-R*, *WR-Зп*, *RD-Чт*, *CS-ВУ*.

Микросхема содержит двунаправленную шину данных и три двунаправленных 8-разрядных канала ввода-вывода: *ШД*, *КА*,

КВ и *КС*. Для организации обмена БИС ППИ использует 6 входных управляющих сигналов.

В состав БИС ППИ входят формирователи шины данных и каналов *ВхФ*, *ВыхФ*, регистр управляющего слова *РУС*, схема выбора каналов *СВК*, регистры каналов *Р1А*, *Р2А*, *РС*, *РВ*, схема управления каналом *СУС*, схема управления регистром канала *КС*, *СУС RC*, схема формирования сброса *СФС*.

Работа БИС ППИ начинается с загрузки *РУС*. Загрузка *РУС* происходит информацией с *ШД* при наличии сигналов *ВУ* = «0», *Зп* = «0», *Чт* = «1», *А1* = *А0* = «1». После этого БИС ППИ переходит в один из трех режимов: 0— простой ввод-вывод, 1 — стробируемый ввод-вывод; 2 — двунаправленный обмен.

Режим работы *КА*, *КВ* и *КС* можно изменить как в начале, так и в процессе выполнения программы командой OUT (см. табл. 3.9).

Режимы *КА* и *КВ* могут быть различными. Режим *КС* (обеих полубайтовых групп) зависит от режимов *КВ* и *КА*.

При каждом изменении режима канала все входные и выходные регистры каналов и триггера состояния сбрасываются в состояние «0».

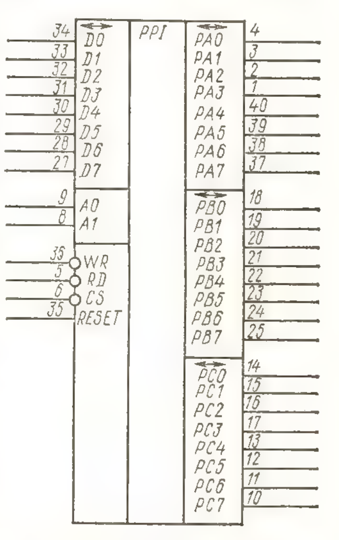


Рис. 3.28. Условное графическое обозначение микросхемы КР580ИК55

Форматы управляющих слов, определяющие режим работы каналов, приведены на рис. 3.30, а графическое представление режимов работы — на рис. 3.31. Каналы *КВ* и *КА* программируются полностью, в то время как *КС* можно программировать по частям. Любой из восьми разрядов регистра канала *КС* может быть установлен программным способом в «1» или «0» в режимах 1 или 0.

Если микросхема запрограммирована в режимы 1 или 2, то в канал *КС* выдаются сигналы, которые могут быть использованы в качестве запросов прерывания работы МП. Запретить или разрешить формирование этих сигналов можно операцией установки (сброса) разрядов в регистре канала *КС*. Это позволяет программисту запретить или разрешить обслуживание любого внешнего устройства.

При работе БИС ППИ в режиме 0 обеспечивается ввод или вывод информации через любой из трех каналов. Сигналов управления (квитирования об установлении связи с периферийным устройством) не требуется; данные просто коммутируются через выбранный канал. В этом режиме микросхему можно представить как два байтовых и два полубайтовых канала ввода-вывода.

В этом режиме выводимая информация запоминается в выходном регистре выбранного канала и хранится там до перезагрузки его новой информацией. Управляющие слова и конфигурации каналов приведены на рис. 3.32 ... 3.35.

При работе БИС ППИ в режиме 1 обеспечивается возможность пересылки информации через выбранный канал по стробирующим сигналам. Данные коммутируются с использованием каналов *КА* и *КВ*. Линии канала *КС* используются для приема и выдачи сигналов управления. Рассмотрим эти управляющие сигналы.

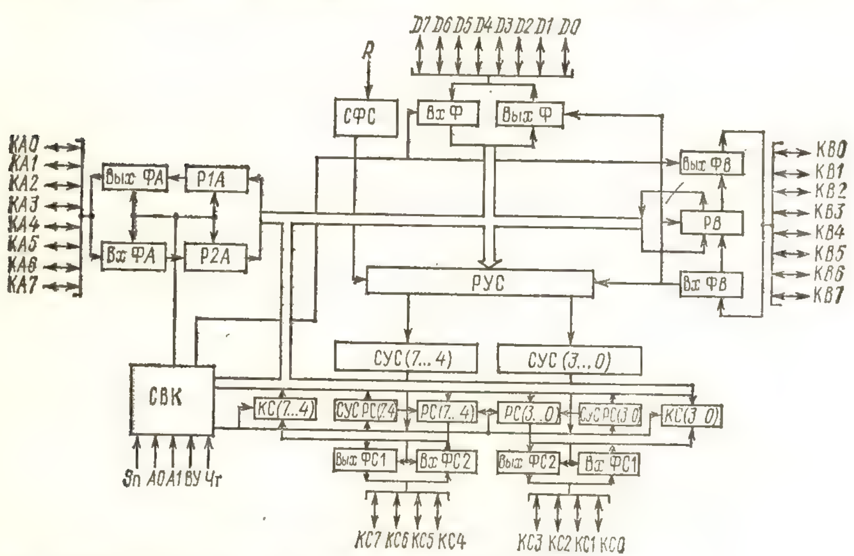


Рис. 3.29. Структурная схема микросхемы КР580ИК55

При подаче на *КС4* (для *КА*) либо *КС2* (для *КВ*) сигнала *Строб приема* (*Стр П*) данные записываются во входной регистр соответствующего канала. Сигнал «1» на выходе *КС5* (для *КА*) либо *КС1* (для *КВ*) — Подтверждение приема (*ПтП*) сигнализирует о том, что входные данные записаны во входной регистр канала. Сигнал *ПтП* устанавливается в «1» с задержкой относительно положительного фронта сигнала *СтрП* и сбрасывается в «0» с задержкой относительно положительного фронта сигнала *Чт* (при переходе сигнала *Чт* из «0» в «1»).

Сигнал *Запрос прерывания* (*ЗПр*)-«1» (*КС3*) может использоваться для прерывания работы МП. Этот сигнал установится в «1», если *СтрП*-«1» и соответствующий разряд канала *КС*, используемый как триггер разрешения прерывания установлен в «1». Сигнал *ЗПр* сбрасывается в «0» после отрицательного фронта сигнала *Чт*. Такая процедура установки (сброса) сигнала *ЗПр* позволяет внешнему устройству ввода простой записью данных в канал микросхемы запрашивать обслуживание МП. Для раз-

Таблица 3.18. **Назначение выводов микросхемы КР580ИК55**

Номер вывода	Назначение	Разряд	Обозначение	Тип
1	Канал А	3	К*А*3	Вход-выход
2	Канал А	2	К*А*2	Вход-выход
3	Канал А	1	К*А*1	Вход-выход
4	Канал А	0	К*А*0	Вход-выход
5	Чтение	—	Чт	Вход
6	Выбор устройства	—	ВУ	Вход
7	Общий	—		—
8	Шина адреса	1	А1	Вход
9	Шина адреса	0	А0	Вход
10	Канал *С*	7	К*С*7	Вход-выход
11	Канал *С*	6	К*С*6	Вход-выход
12	Канал *С*	5	К*С*5	Вход-выход
13	Канал *С*	4	К*С*4	Вход-выход
14	Канал *С*	0	К*С*0	Вход-выход
15	Канал *С*	1	К*С*1	Вход-выход
16	Канал *С*	2	К*С*2	Вход-выход
17	Канал *С*	3	К*С*3	Вход-выход
18	Канал *В*	0	К*В*0	Вход-выход
19	Канал *В*	1	К*В*1	Вход-выход
20	Канал *В*	2	К*В*2	Вход-выход
21	Канал *В*	3	К*В*3	Вход-выход
22	Канал *В*	4	К*В*4	Вход-выход
23	Канал *В*	5	К*В*5	Вход-выход
24	Канал *В*	6	К*В*6	Вход-выход
25	Канал *В*	7	К*В*7	Вход-выход
26	Питание	—	U_{cc}	—
27	Шина данных	7	*D*7	Вход-выход
28	Шина данных	6	*D*6	Вход-выход
29	Шина данных	5	*D*5	Вход-выход
30	Шина данных	4	*D*4	Вход-выход
31	Шина данных	3	*D*3	Вход-выход
32	Шина данных	2	*D*2	Вход-выход
33	Шина данных	1	*D*1	Вход-выход
34	Шина данных	0	*D*0	Вход-выход
35	Сброс	—	*R*	Вход
36	Запись	—	Зп	Вход
37	Канал *А*	7	К*А*7	Вход-выход
38	Канал *А*	6	К*А*6	Вход-выход
39	Канал *А*	5	К*А*5	Вход-выход
40	Канал *А*	4	К*А*4	Вход-выход

Примечание. Каналы *А*, *В* и *С*, а также шины данных трехстабильные.

Режим 1 (ввод информации)

7	6	5	4	3	2	1	0
Ввод — вывод	Ввод — вывод	Подтверждение приема	Разрешение прерывания	Запрос прерывания	Разрешение прерывания	Подтверждение приема	Запрос прерывания
Канал A					Канал B		

Режим 1 (вывод информации)

7	6	5	4	3	2	1	0
Строб записи	Разрешение прерывания	Ввод — вывод	Ввод — вывод	Запрос прерывания	Разрешение прерывания	Строб записи	Запрос прерывания
Канал A					Канал B		

Режим 2

7	6	5	4	3	2	1	0
Строб записи	Разрешение прерывания по выводу	Подтверждение приема	Разрешение прерывания по вводу	Запрос прерывания			
Канал A					Канал B		

Рис. 3.30. Формат управляющего слова

решения формирования сигнала *ЗПр* по каналу *КА* используется разряд *КС4*, а по каналу *КВ* — разряд *КС2*. Разряды *КС6* и *КС7* могут использоваться как для ввода, так и для вывода данных. Если они запрограммированы для ввода, то прочитать информацию на них можно обычным чтением канала *КС*. Если же эти разряды запрограммированы на вывод, то для передачи информации ис-

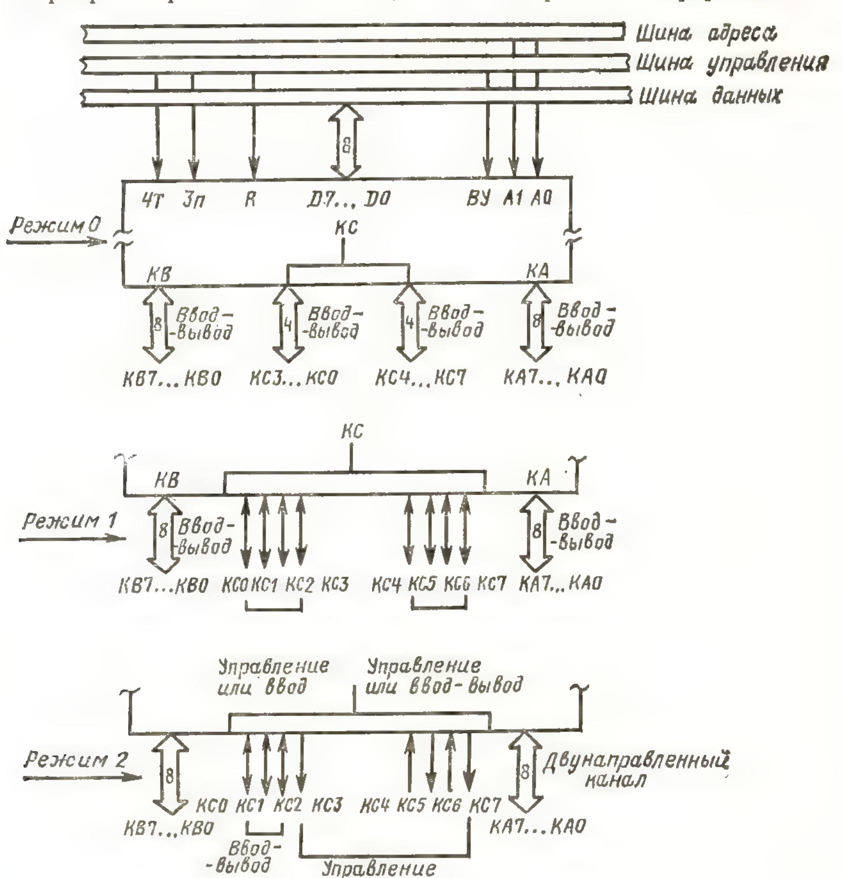


Рис. 3.31. Графическое представление режимов

пользуется режим установки (сброса) разрядов канала *КС*. Временна́я диаграмма ввода данных в режиме «0» приведена на рис. 3.36.

Сигналы управления при выводе данных имеют следующее назначение.

Сигнал *Строб записи* (*СтрЗп*) = «0» свидетельствует о том, что МП произвел запись данных в выходной регистр *ШД*.

Сигнал *СтрЗп* переходит в «0» после положительного фронта сигнала *Зп*, а в «1» после отрицательного фронта сигнала *Под-*

тверждение записи (*ПтЗп*). Сигнал *ПтЗп* = «0» свидетельствует о том, что внешнее устройство приняло данные.

Сигнал *ЗПр* = «1» при выводе данных появляется при *СтрЗп* = *ПтЗп* = «1» и соответствующий разряд канала *КС*, используемый как триггер разрешения прерывания по данному каналу, установлен в «1». Сброс сигнала *ЗПр* происходит после отрицательного фронта сигнала *Зп*.

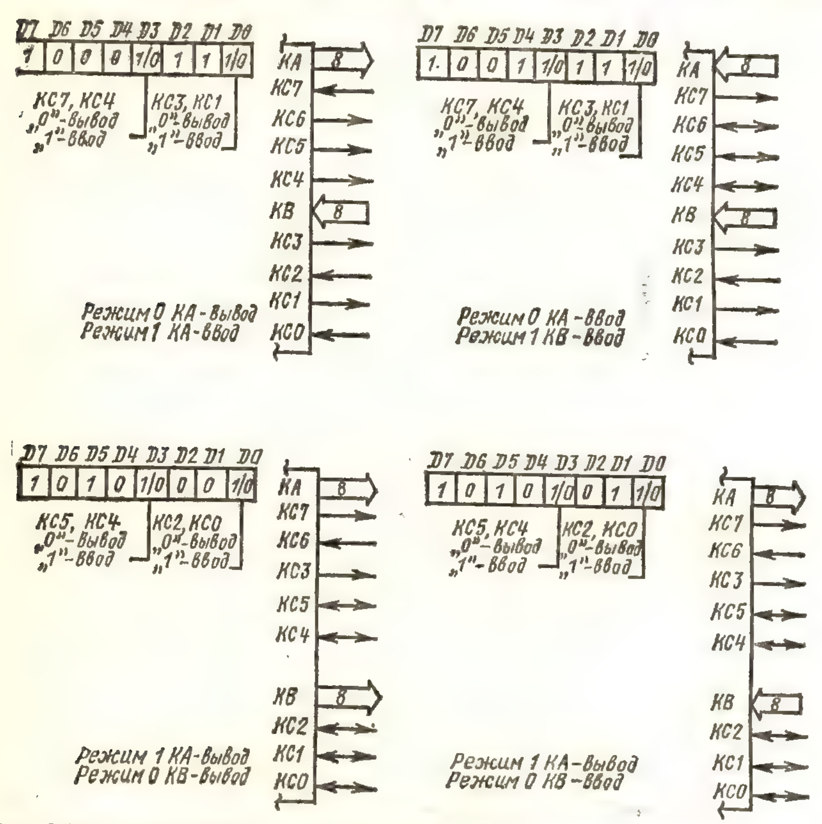


Рис. 3.32. Управляющие слова и конфигурация каналов в режимах 0 и 1

Для разрешения формирования сигнала *ЗПр* при выводе данных по каналу *КА* используется разряд *КС6*, а по каналу *КВ* — *КС2*.

Разряды *КС4* и *КС5* могут использоваться как для ввода, так и для вывода данных. Доступ к ним тот же, что и для ввода (см. выше). После перевода ППИ в режим 1 для вывода данных сигналы *СтрЗп* и *ЗПр* = «0». Перевести сигнал *СтрЗп* в «1» можно либо подачей сигнала *ПтЗп* = «0», либо установкой разрядов *КС7* (для *КА*) и *КС1* (для *КВ*) в «1». Затем следует устано-

вить в «1» соответствующий триггер разрешения прерывания, при этом *ЗПр* перейдет в «1». После этого ППИ готов к выводу данных в режиме 1.

Каналы *A* и *B* независимо один от другого могут быть запрограммированы на ввод или вывод в режиме 1 (рис. 3.33 ... 3.39).

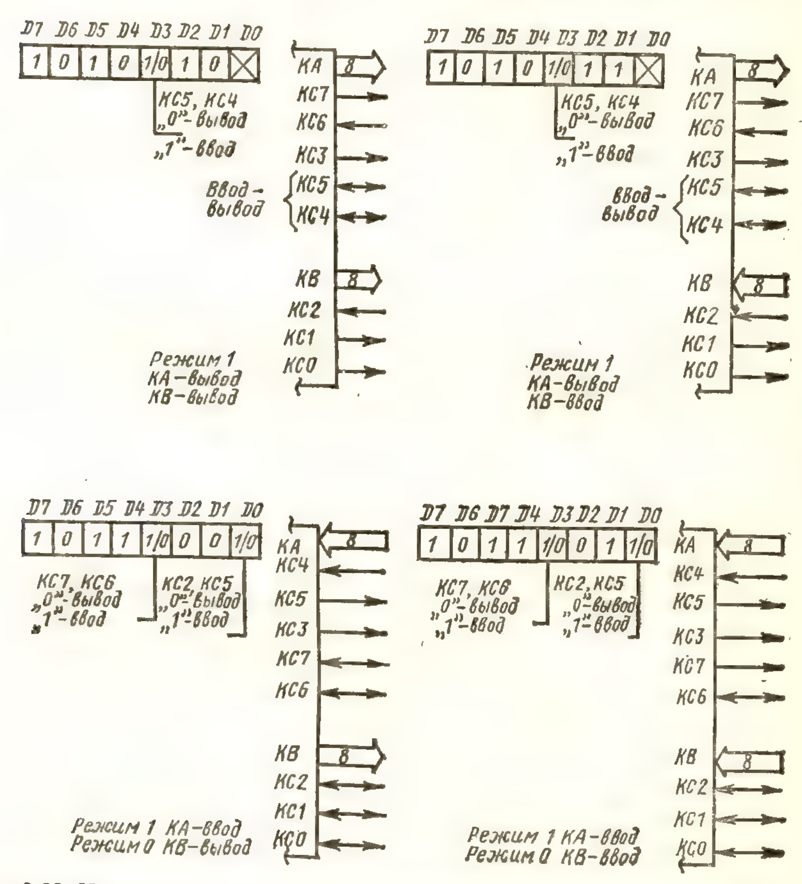


Рис. 3.33. Управляющие слова и конфигурация каналов в режимах 1, 1 и 0

При работе микросхемы в режиме 2 возможен обмен информацией с периферийными устройствами и схемами по 8-разрядному двунаправленному трехстабильному каналу ввода-вывода данных. Для простого обмена, аналогичного протоколу режима 1, используют пять линий канала *КС*.

В режиме 2 может работать только канал *КА*, причем входные и выходные данные фиксируются во входном и выходном регистрах соответственно.

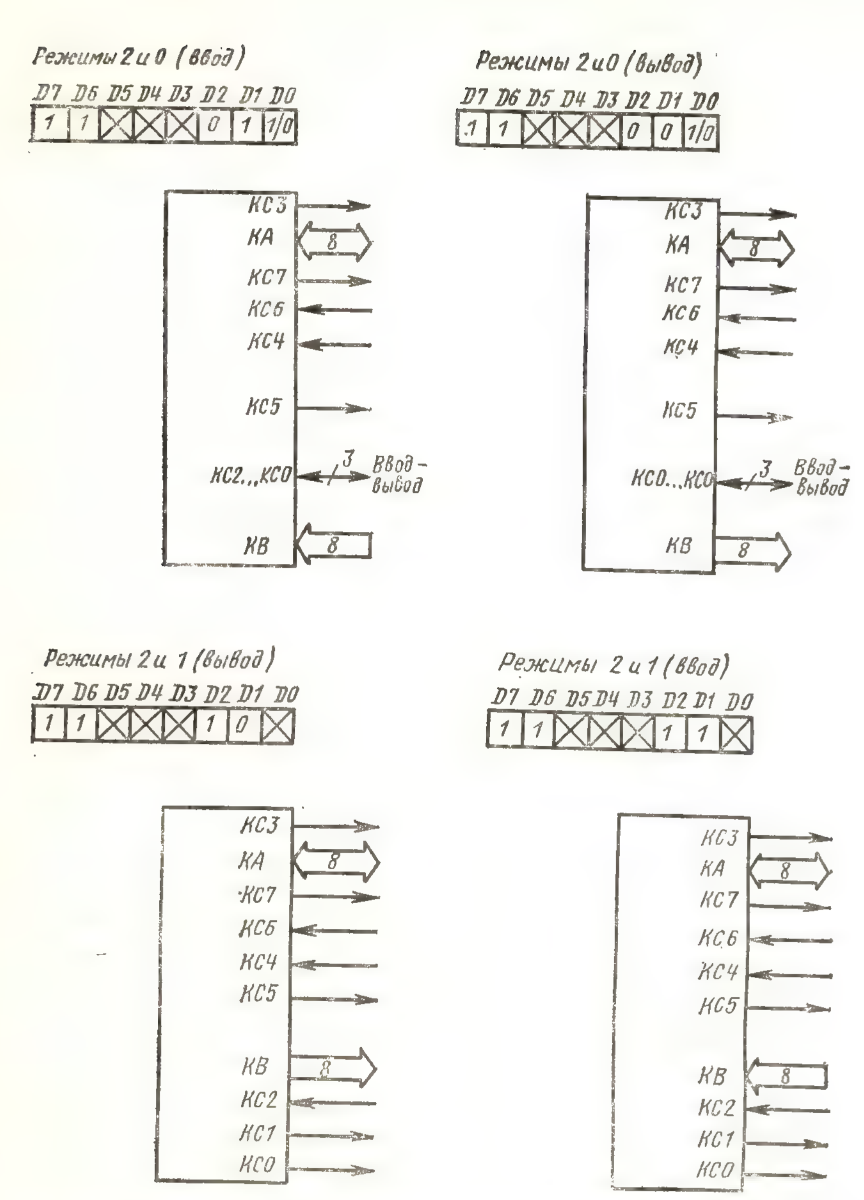


Рис. 3.34. Управляющие слова и конфигурация каналов в режимах 0 и 2, 1 и 2

D7 D6 D5 D4 D3 D2 D1 D0
1 0 0 0 1/0 1 0 1/0
КС7, КС4 Вывод - „0" Ввод - „1"
КС3 „0"-Вывод „1"-Ввод
КА 8 КС7 КС6 КС5 КС4 КВ 8 КС3 КС2 КС1 КС0
Режим 0 КА - Вывод
Режим 1 КА - Вывод

D7 D6 D5 D4 D3 D2 D1 D0
1 0 0 1 1/0 1 0 1/0
КС7, КС4 Вывод - „0" Ввод - „1"
КС3 „0"-Вывод „1"-Ввод
КА 8 КС7 КС6 КС5 КС4 КВ 8 КС3 КС2 КС1 КС0
Режим 0 КА - Ввод
Режим 1 КВ - Вывод

D7 D6 D5 D4 D3 D2 D1 D0
1 0 1 1 1/0 1 0 ☒
КС7, КС6 „0"-Вывод „1"-Ввод
КА 8 КС5 КС4 КС3 КС7 КС6 Ввод - вывод КВ 8 КС2 КС1 КС0
Режим 1
КА - Ввод
КВ - Вывод

D7 D6 D5 D4 D3 D2 D1 D0
1 0 1 1 1/0 1 1 ☒
КС7, КС6 „0"-Вывод „1"-Ввод
КА 8 КС5 КС4 КС3 КС7 КС6 Ввод - вывод КВ 8 КС2 КС1 КС0
Режим 1
КА - Ввод
КВ - Ввод

Рис. 3.35. Управляющие слова и конфигурация каналов в режимах 0 и 1 и режиме 1

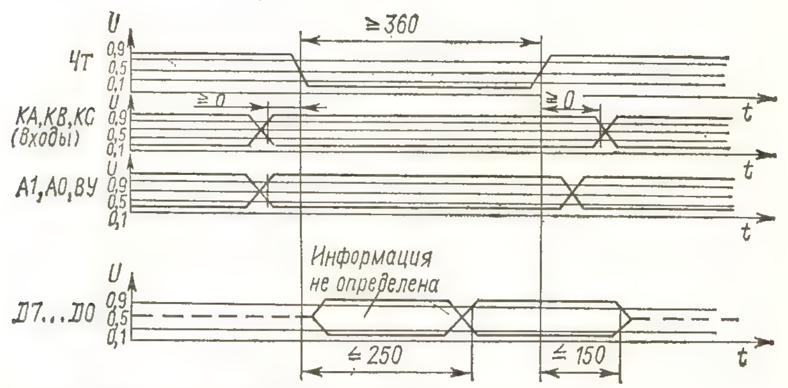


Рис. 3.36. Временна́я диаграмма ввода информации в режиме 0

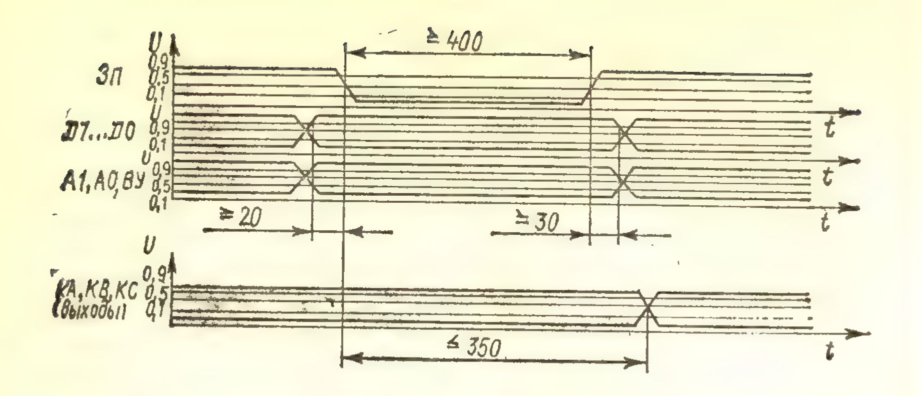


Рис. 3.37. Временна́я диаграмма вывода информации в режиме 0

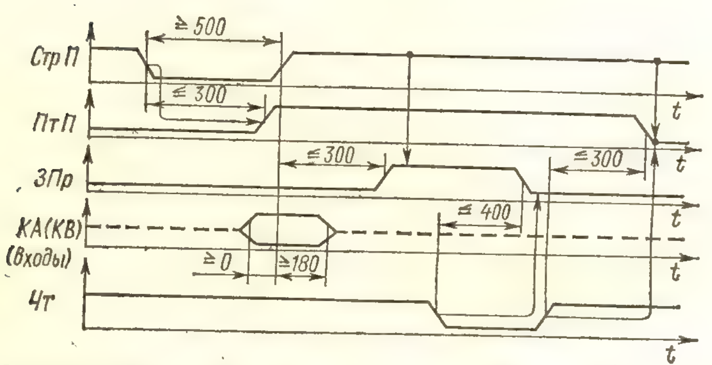


Рис. 3.38. Временна́я диаграмма ввода данных в режиме 1

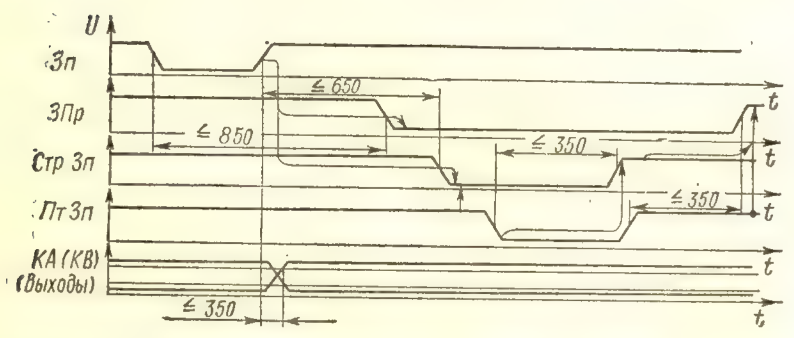


Рис. 3.39. Временна́я диаграмма вывода данных в режиме 1

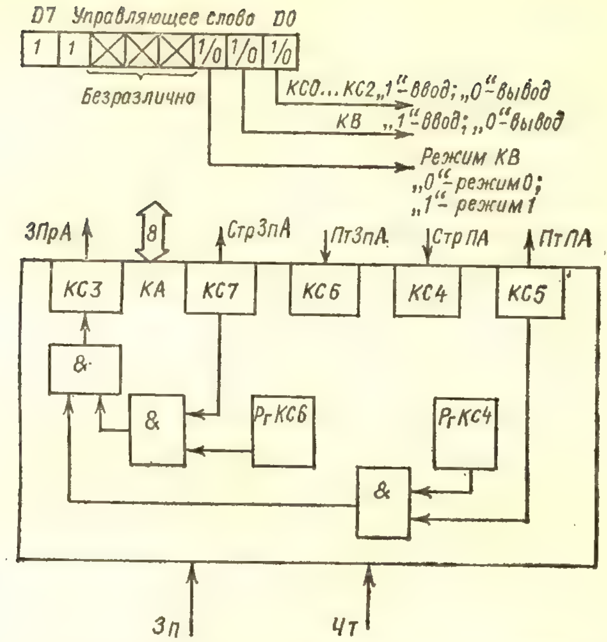


Рис. 3.40. Формат управляющего слова и конфигурация каналов в режиме 2

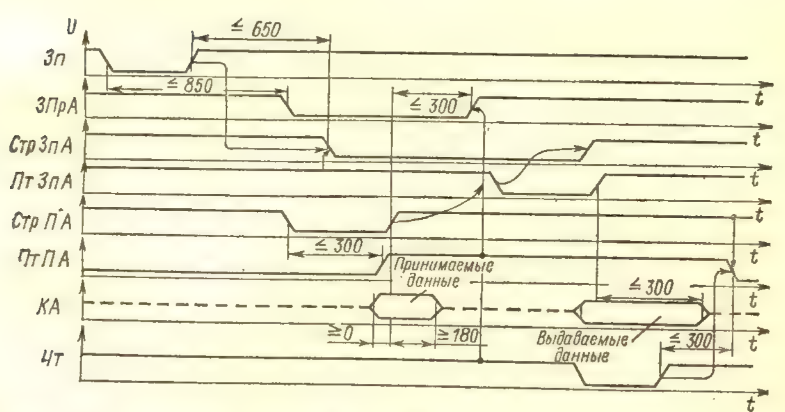


Рис. 3.41. Временная диаграмма работы микросхемы в режиме 2

Таблица 3.19. Статические параметры микросхемы КР580ИК55

Параметр, единица измерения	Обозначение	Значения		Режим измерения
		мин.	макс.	
Выходное напряжение «0» по ШД, В	U_{OLD}	—	0,45	$I_{OLD} = 2,5$ мА
Выходное напряжение «0» по каналу КА, КВ, КС, В	U_{OLK}	—	0,45	$I_{OLK} = 1,7$ мА
Выходное напряжение «1» по ШД, В	U_{OHD}	2,4	—	$I_{OHD} = -0,4$ мА
Выходное напряжение «1» по каналам КА, КВ, КС, В	U_{OHK}	2,4	—	$I_{OHK} = -0,2$ мА
Входное напряжение «1», В	U	2,2	—	
Входное напряжение «0», В	U_{IL}	−0,5	0,8	—
Ток потребления, мА	I_{CG}	—	120	—
Ток утечки на управляющих входах, мкА	I_{IZ}	−10	10	—
Выходной ток в состоянии «выключено», мкА	I_{OZ}	−10	10	—
Выходной ток высокого уровня по каналам КВ и КС, мА	I_{OHK}	1	4	$U_{CG} = 4,5$ В; $R_L = 750$ Ом.

Формат управляющего слова и функциональная схема ввода-вывода данных приведены на рис. 3.40, временна́я диаграмма работы канала *КА* в режиме 2 — на рис. 3.41.

Ниже приведено распределение сигналов управления вводом-выводом в режиме 2 (см. рис. 3.30, 3.40 и 3.41). Сигнал «1» на выходе *ЗПрА* (*КС3*) может быть использован для прерывания работы МП при операциях как ввода, так и вывода данных. В состояние «0» сигнал переходит после отрицательного фронта сигнала *Зп* (см. рис. 3.39, 3.41). В состояние «1» сигнал переходит после положительного фронта сигнала *Строб приема А* (*СтрПА*), если сигналы *Подтверждение приема А* (*ПтПА*) и *Чт* в состоянии «1» (см. рис. 3.41 применительно к каналу *КА*).

Сигналы *СтрПА* и *ПтПА* соответствуют разрядам *КС4* и *КС5*. При *СтрПА* = «0» данные записываются во входной регистр канала *КА*. Сигнал *ПтПА* = «1» свидетельствует о том, что операция записи завершена. Разряд *КС4* используется для разрешения формирования сигнала *ЗПрА*.

Сигнал *СтрЗпА* = «0» подтверждает окончание записи в выходной регистр канала *КА*. Сброс сигнала в «1» происходит после отрицательного фронта сигнала *ПтЗпА* (см. рис. 3.41).

Для разрешения выработки сигнала *ЗПрА* при выводе информации из канала *КА* используется разряд *КС6*. Следует особо отметить назначение сигнала *R*. При подаче этого сигнала каналы *КА*, *КВ* и *КС* переходят в третье состояние. Все каналы настраиваются на режим 0 для ввода информации. При снятии сигнала *R* ППИ не меняет своего состояния. Статические параметры ППИ приведены в табл. 3.19.

3.6. МИКРОСХЕМА К580ИК57 (КР580ИК57)

Микросхема представляет собой БИС четырехканального программируемого контроллера прямого доступа к памяти (ПДП). Каждый из четырех каналов адресует область внешней памяти путем инкрементирования выбранного адреса. ПДП имеет приоритетную логику, реализующую запросы от четырех периферийных устройств и производит счет циклов прямого доступа к памяти каждого канала.

Использование БИС ПДП позволяет существенно сократить аппаратные затраты при реализации прямого доступа к памяти.

Условное графическое обозначение БИС ПДП приведено на рис. 3.42, назначение выводов — в табл. 3.20.

Рис. 3.42. Условное графическое обозначение микросхемы КР580ИК57

На рис. 3.43 приведена структурная схема микросхемы К580ИК57. Далее даны соответствия латинских и отечественных обозначений сигналов и выводов микросхемы КР580ИК57: *I/O R* — Чт В/В, *I/O W* — Зп В/В, *MEMR* — ЧтП, *MEMW* — ЗпП, *MARK* — М128, *READY* — Гт, *HLDA* — ПЗхв, *ADSTB* — СтрА, *AEN* — РА, *HRQ* — ЗЗхв, *CS* — ВМ, *CLK* — ТИ, *RESET* — Уст, *DACK* — ППд, *DRQ* — ЗПД, *TC* — КС.

Каждый канал ПДП имеет регистр адреса и регистр количества циклов (оба 16-разрядные), шину адреса (*ША*), шину данных (*ШД*). *ША* разделена на две части. *A3* ... *A0* — двунаправленная часть ША, при работе которой во входном режиме указанные разряды адреса используют для инициализации определенного канала, а в выходном режиме — как младший полубайт адреса внешней памяти. *A7* ... *A4* являются выходами и используются для выдачи адреса внешней памяти.

Шина данных (*D7* ... *D0*) обеспечивает двунаправленный обмен информацией между МП и ПДП. По этой шине принимаются управляющие слова и выдается старший байт адреса внешней памяти.

Сигнал *Чт В/В* разрешает (во входном режиме) чтение 8-разрядного регистра состояния или записанных в ЗУ ПДП начального адреса и числа циклов ПД любого из каналов. В выходном режиме сигнал *Чт В/В* разрешает выдачу информации из внешнего устройства ввода-вывода. Сигнал *Зп В/В* разрешает (во входном режиме) загрузку регистров установки режимов, началь-

Таблица 3.20. **Назначение выводов микросхемы КР580ИК57**

Номер вывода	Назначение	Разряд	Обозначение	Тип
1	Чтение ввода-вывода	—	Чт В/В	Вход-выход
2	Запись ввода-вывода	—	Зп В/В	
3	Чтение памяти	—	ЧтП	Выход
4	Запись в память	—	ЗпП	
5	Модуль 128	—	М128	
6	Готовность	—	Гт	
7	Подтверждение захвата	—	ПЗхв	Вход
8	Строб адреса	—	СтрА	
9	Разрешение адреса	—	РА	
10	Запрос захвата	—	ЗЗхв	
11	Выбор микросхемы	—	ВМ	
12	Тактовый импульс	—	ТИ	
13	Установка	—	Уст	
14	Подтверждение ПД канала 2	—	ППД2	Выход
15	Подтверждение ПД канала 3	—	ППД3	
16	Запрос прямого доступа каналов 0....3	—	ЗПД3	Вход
17		—	ЗПД2	
18		—	ЗПД1	
19		—	ЗПД0	
20	Общий	—	*GND*	—
21	Шина данных	7	*D*7	Вход-выход
22		6	*D*6	
23		5	*D*5	
24	Подтверждение ПД канала 1		ППД1	Выход
25	Подтверждение канала 0		ППД0	
26	Шина данных	4	*D*4	Вход-выход
27		3	*D*3	
28		2	*D*2	
29		1	*D*1	
30		0	*D*0	
31	Питание	—	U_{CC}	—
32	Шина адреса	0	*A*0	Вход-выход
33		1	*A*1	
34		2	*A*2	
35		3	*A*3	
36	Конец счета	—	КС	
37	Шина адреса	4	*A*4	Выход
38		5	*A*5	
39		6	*A*6	
40		7	*A*7	

Примечание. Выводы чтения, записи, а также шины данных и адреса трехстабильные.

ного адреса значения количества циклов для любого канала ПДП. В выходном режиме сигнал *Зп В/В* разрешает запись информации во внешнее устройство ввода-вывода.

Выходные сигналы *ЧтП* и *ЗпП* предназначены для внешнего ЗУ.

Входные сигналы *ЗПД3 ... ЗПД0* поступают асинхронно из внешних устройств и воспринимаются микросхемой как запросы на обмен с ЗУ.

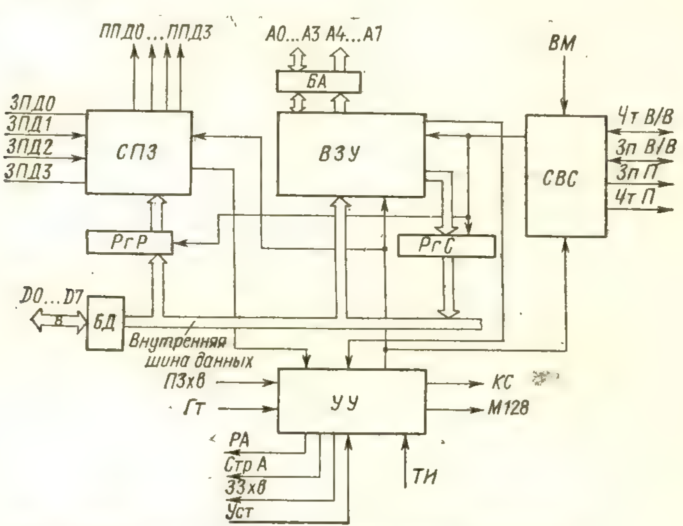


Рис. 3.43. Структурная схема микросхемы КР580ИК57

Выходные сигналы *ППД*3 ... *ППД*0 являются ответными по отношению к *ЗПД*3 ... *ЗПД*0. Они вырабатываются микросхемой в соответствии с приоритетами внешних устройств.

Выходной сигнал *КС* вырабатывается при установке в нуль 14-разрядного регистра количества циклов и указывает периферийным устройствам, что данный цикл ПД последний.

Выходной сигнал *М128* появляется в каждом 128-м цикле от конца массива, а также во время действия сигнала *КС*.

Схема приема запросов *СПЗ* предназначена для приема и привязки несинхронных сигналов запросов на организацию прямого доступа к памяти от четырех устройств, а также выдачи ответных сигналов подтверждения. Каждый из четырех каналов связан с БИС ПДП отдельными линиями запросов и подтверждения прямого доступа. Выдача ответного сигнала подтверждения для соответствующего канала происходит в зависимости от его приоритета.

Внутреннее запоминающее устройство микросхемы ВЗУ предназначено для хранения начального адреса и числа циклов ПД для каждого канала в 16-разрядных регистрах адреса *РгА* и цик-

Таблица 3.21. **Виды обмена данными при ПД**

Разряд 15	Разряд 14	Вид обмена
0	0	Цикл проверки ПД
0	1	Цикл записи ПД
1	0	Цикл чтения ПД
1	1	Запрещенная комбинация

лов *РгЦ* соответственно. *РгА* загружается адресом первой ячейки памяти, к которой должно быть обращение. Младшие 14 разрядов *РгЦ* указывают число циклов ПД (минус один) до конца счета (до появления сигнала *КС*). Разряды 14-й и 15-й *РгЦ* указывают на вид обмена данными при ПД (табл. 3.21).

Выходной сигнал *ЗЗхв* запрашивает у МП разрешение на управление системными шинами (линиями).

Выходной сигнал *ПЗхв* является ответом МП на сигнал *ЗЗхв*. При появлении *ПЗхв* системные шины (линии) свободны.

Входной сигнал *Гт* предназначен для обеспечения совместной работы ПДП и медленных внешних устройств. Сигнал может поступать асинхронно. Он отражает готовность внешнего устройства к ведению обмена.

Выходной сигнал *СтрА* указывает, что на шине данных выдан старший байт адреса внешнего ЗУ.

Выходной сигнал *РА* используется для блокировки адресных шин в невыбранных устройствах.

Входной сигнал *ВМ* позволяет активизировать данную БИС.

Входной сигнал *Уст* предназначен для установки схемы в исходное состояние.

Входной сигнал *ТИ* обеспечивает функционирование микросхемы. Обычно этим сигналом является сигнал *Ф2* микропроцессора КР580ИК80А. БИС ПДП имеет один номинал напряжения питания +5 В.

При реализации циклов прямого доступа к памяти БИС ПДП функционирует в следующих режимах: автозагрузка, конец счета — стоп, удлиненная запись, обычная запись, циклический сдвиг приоритетов, фиксированный приоритет, маскирование, чтение, запись, проверка.

В блоке ВЗУ происходит формирование массива адресов инкрементированием текущего адреса. Младший байт адреса *А7 ... А0* помещается в буфер адреса *БА*, старший байт (*А15 ... А8*) — на буфер данных *БД* (рис. 3.43).

Последовательностью операций в течение циклов ПД управляет устройство управления *УУ*, которое принимает, вырабатывает и передает следующие сигналы:

ЗЗхв — запрашивает управление системными шинами;

ПЗхв — ответ на сигнал *ЗЗхв*;

ТИ — тактовые импульсы *Ф2* ТТЛ-уровня;

СтрА — стробирует старший байт адреса на шине данных;

РА — указывает, что протекают циклы ПД;

КС — указывает выбранному устройству, что текущий цикл ПД может быть последним для данного массива при условии установки *КС*-стоп в состояние «1»;

М128 — указывает выбранному устройству, что текущий цикл ПД является 128-м от конца массива данных;

Гт — для «удлинения» циклов обращения к памяти путем введения микросхемы в состояние ожидания готовности;

Уст — для сброса регистра установки режимов, что приводит к запрету работы всех каналов во всех режимах.

Буфер адреса БА (см. рис. 3.43) предназначен для приема и выдачи адреса памяти либо одного из внутренних регистров схемы. *БА* разделен на две части. Адресные линии *А3 ... А0* в состоянии программирования указывают номер регистра, инициализированного для обмена. При обслуживании циклов ПД эти линии являются входными и по ним передаются четыре младших разряда адреса памяти.

Адресные линии *А7 ... А4* являются всегда выходными. Информация на них соответствует разрядам генерируемого адреса памяти.

Схема выработки сигналов *Запись-Чтение СВС* осуществляет прием, формирование и выдачу сигналов, обеспечивающих обмен информаций между процессором и микросхемой — с одной стороны, и памятью и периферийными устройствами — с другой. *СВС* принимает и выдает следующие сигналы:

Зп В/В — двунаправленный сигнал, позволяющий в режиме программирования загрузить содержимое шины данных во внутренний регистр режимов, в регистр адреса или в регистр количества циклов; при обслуживании внешних устройств сигнал разрешает запись данных в периферийное устройство во время циклов ПД;

ЧтП — для чтения данных из адресуемых ячеек памяти в течение циклов чтения ПД;

ЗпП — для записи данных в адресуемые ячейки памяти в течение циклов записи ПД;

ВМ — разрешает прием внешних сигналов *Чт В/В* и *Зп В/В* в состоянии программирования;

Чт В/В — для считывания данных из внешнего устройства при обслуживании ПД.

В зависимости от цикла ПД (чтение или запись) схема вырабатывает парные сигналы. При чтении генерируются сигналы *ЧтП* и *Зп В/В*, при записи — *ЗпП* и *Чт В/В*.

Буфер данных БД представляет собой 8-разрядное устройство, обеспечивающее двунаправленный обмен информацией между БИС и системной шиной данных. Информация, поступающая на *БД* с системной шины данных, передается в регистр установки

режимов либо в ЗУ. Информацией, поступающей с внутренней шины данных на *ШД*, могут являться содержимое регистра адреса, регистра количества циклов, регистра состояния БИС. В течение циклов ПД выдаются старшие восемь разрядов адреса памяти.

Регистр установки режимов *РгР* хранит информацию о режимах работы БИС, к которым относятся «Автозагрузка», «Конец счета-стоп», «Удлиненная запись», «Обычная запись», «Циклический сдвиг приоритета» и «Фиксированный приоритет».

Разряды *РгР* показаны на рис. 3.44.

РгР обычно загружается после установки *РгА* и *РцЦ* и сбрасывается подачей сигнала *Уст*.

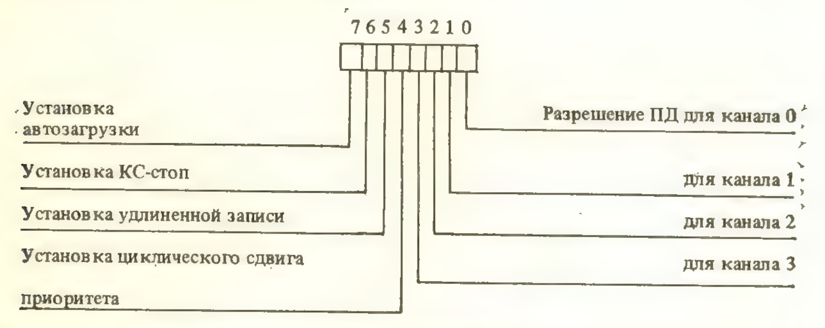


Рис. 3.44. Формат регистра установки режимов

Разряды 3 ... 0 *РгР* разрешают работу соответствующего канала. Разряды 7 ... 4 обеспечивают соответствующий режим работы БИС. Так, при записи «1» в разряд 4 *РгР* приоритет каждого канала изменяется.

Обслуженный канал будет иметь самый низкий приоритет. Порядок обслуживания каналов будет установлен в соответствии с их номерами $0 \to 1 \to 2 \to 3 \to 0$. Если разряд 4 *РгР* установлен в «0», то каждый канал будет иметь фиксированный приоритет. Так, канал 0 имеет наивысший приоритет, а канал 3 — самый низкий. При записи «1» в разряд 5 *РгР* устанавливается режим «Удлиненная запись». В этом режиме продолжительность сигналов *ЗпП* и *Зп В/В* увеличивается при отсутствии сигнала готовности внешнего устройства. При этом БИС входит в состояние ожидания.

При записи «1» в разряд 6 *РгР* устанавливается режим «Конец счета — стоп». В этом случае после появления сигнала *КС* обслуженный канал окажется запрещенным. Если необходимо продолжить обслуживание данного канала, перепрограммируют разряд разрешения данного канала. При «0» в разряде *РгР* появление сигнала *КС* не запрещает повторное обслуживание канала.

При «1» в разряде 7 *РгР* устанавливается режим «Автозагрузка», позволяющий каналу 2 повторно пропустить массив данных или связать ряд массивов без программного вмешательства.

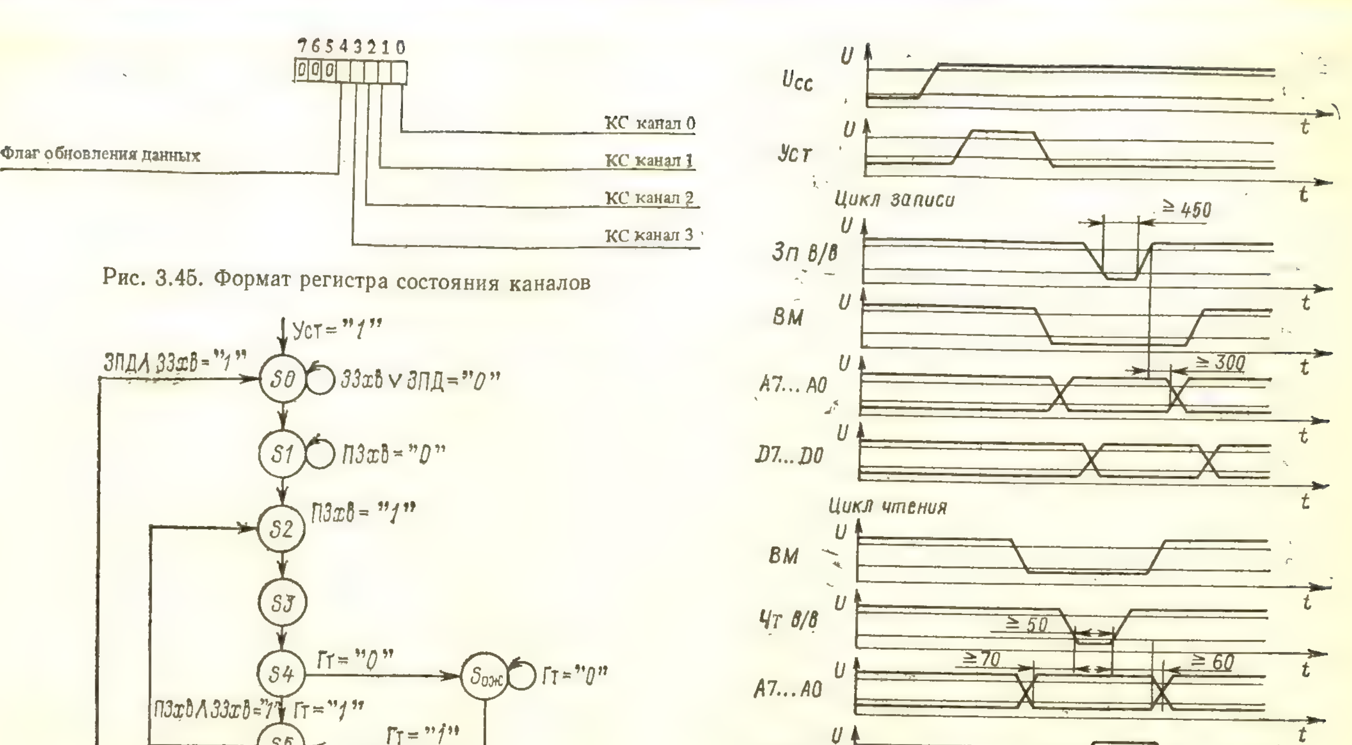


Рис. 3.45. Формат регистра состояния каналов

Рис. 3.46. Диаграмма работы микросхемы КР580ИК57

Рис. 3.47. Временна́я диаграмма выполнения циклов записи и чтения

Таблица 3.22. **Адресация внутренних регистров каналов ПД**

Операция	Управляющие сигналы			
	ВМ	ЗпВ/В	ЧтВ/В	А3
Запись в регистр канала	0	0	1	0
Чтение из регистра канала	0	1	0	0
Запись в РгР	0	0	1	1
Чтение РгС	0	1	0	1

Регистры канала 2 устанавливаются для передачи первого массива. Регистры канала 3 хранят информацию для переустановки регистров канала 2. После передачи первого массива и появления сигнала *КС* содержимое регистров канала 3 передается в соответствующие регистры канала 2. Каждый раз, когда в регистрах канала 2 происходит «подмена» данных информацией регистров канала 3, устанавливается разряд «Флаг обновления данных» в регистре состояния каналов.

Регистр состояния каналов РгС указывает номер канала, который достиг конца счета. Кроме того, в *РгС* входит разряд «Флаг обновления данных», описанный выше. Разряды *РгС* показаны на рис. 3.45.

В процессе функционирования в составе микропроцессорной системы микросхема может находиться в одном из следующих состояний: исходном, программирования, ожидания, обслуживания.

В исходное состояние микросхему переводит внешний сигнал *Уст*. В этом состоянии маскируются все запросы каналов *ПД*, а буферы *А3 ... А0* переводятся в состояние приема информации. В состоянии программирования МП имеет доступ к внутренним регистрам выбранного канала в соответствии с табл. 3.22 и 3.23.

В состоянии ожидания микросхема находится либо от момента окончания программирования до выдачи сигнала *ППД*, либо в промежутках между циклами *ПД*. На рис. 3.46 приведена диаграмма ожидания и работы БИС ПДП.

После получения от МП сигнала *ПЗхв* при наличии сигнала запроса микросхема вырабатывает сигнал *ППД* и переходит в состояние обслуживания, в котором системные шины находятся под управлением БИС ПДП.

Временна́я диаграмма работы БИС приведена на рис. 3.47.

Статические параметры микросхемы КР580ИК57 приведены в табл. 3.24.

Таблица 3.23. Коды выборки регистров

Регистр	Байт	Адресные входы				Двунаправленная шина данных							
		A3	A2	A1	A0	7	6	5	4	3	2	1	0
Адрес *КАН0*	МЛ СТ	0 0	0 0	0 0	0 0	A7 A15	A6 A14	A5 A13	A4 A12	A3 A11	A2 A10	A1 A9	A0 A8
Количество циклов *КАН0*	МЛ СТ	0 0	0 0	0 0	1 0	C7 Чт	C6 Зп	C5 C13	C4 C12	C3 C11	C2 C10	C1 C9	C0 C8
Адрес *КАН1*	МЛ СТ	0 0	0 0	1 1	0 0								
Количество циклов *КАН1*	МЛ СТ	0 0	0 0	1 1	1 1								
Адрес *КАН2*	МЛ СТ	0 0	1 1	0 0	0 0	A7	A6	A5	A4	A3	A2	A1	A0 — младший байт
Количество циклов *КАН2*	МЛ СТ	0 0	1 1	0 0	1 1	A15	A14	A13	A12	A11	A10	A9	A9 — старший байт
Адрес *КАН3*	МЛ СТ	0 0	1 1	1 1	0 0								
Количество циклов *КАН3*	МЛ СТ	0 0	1 1	1 1	1 1								
Установка режима (только записывается)	—	1	0	0	0	АЗ	КС-стоп	УЗ	ЦПС	РК3	РК2	РК1	РК0
Состояния (только читается)	—	1	0	0	0	0	0	0	ФОД	КС3	КС2	КС1	КС0

Примечание. АЗ — автозагрузка; УЗ — удлиненная запись; ЦПС — циклический сдвиг приоритета; РК — разрешение канала; ФОД — флаг обновления данных; АО ... А15 — начальный адрес; С0 ... С3 — количество циклов.

Таблица 3.24. **Статические параметры микросхемы К580ИК57**

Параметр, единица измерения	Обозначение	Значения		Режим изменения
		мин.	макс.	
Входное напряжение «0», В	U_{IL}	—	0,45	—
Входное напряжение «1», В	U_{IH}	2,4	—	—
Выходное напряжение «0», В	U_{OL}	—	0,45	При токе нагрузки 1,6 мА
Выходное напряжение «1», В	U_{OH}	2,4	—	При токе нагрузки 150 мкА
Напряжение источника питания В	U_{CC}	4,75	5,25	—
Ток, потребляемый от источника питания, мА	I_{CC}	—	120	—
Входной ток утечки, мкА	I_{IZ}	—	50	При $U_{IH} = U_{CC}$
Ток утечки на управляющих выходах, мкА	I_{IZ1}	—	50	—
Входная емкость, пФ	C_I	—	10	$f = 1$ МГц
Емкость нагрузки по неуправляющим выводам, пФ	C_{L2}	—	50	—
Емкость нагрузки по управляющим выводам, пФ	C_{L1}	—	75	—

3.7. МИКРОСХЕМА КР580ВН59

Микросхема представляет собой БИС программируемого контроллера прерываний (БИС ПКП), совместима с МП этой серии и позволяет без введения дополнительных аппаратурных затрат вырабатывать до восьми векторов прерывания. Возможно каскадное соединение нескольких БИС ПКП для обеспечения 64 векторов прерывания. БИС ПКП может быть связана с системной магистралью по *шине данных* (*ШД*). *ШД* (8-разрядная) обеспечивает двунаправленный обмен информацией между БИС ПКП и активным устройством.

БИС ПКП имеет 8 входов для приема сигналов запросов прерываний.

Микросхема может вырабатывать трехбайтовую команду CALL и выдавать ее на *ШД*.

Условное графическое обозначение микросхемы приведено на рис. 3.48. Структурная схема БИС ПКП представлена на рис. 3.49. В табл. 3.25 даны назначения выводов. Ниже указаны латинские и отечественные обозначения выводов и сигналов микросхемы КР580ВН59.

CS — ВМ, *WR* — Зп, *RD* — Чт, *CAS* — Кас, *INTA* — ППр, *SP* — ВВдм, *IR* — ЗПр, *INT* — Пр.

БИС ПКП состоит из следующих функциональных блоков:

схема логики чтения-записи *Л Чт/Зп*;

регистр слов команд инициализации *РСКИ*;

схема формирования команды CALL — *ФCALL*;

устройство управления *УУ*;

буфер шины данных *БД*;

регистр слов команд операций *РСКО*;

буфер каскадирования *БКас*;

регистр обслуживания запросов *РОЗПр*;

схема маскирования запросов прерывания и анализа их по приоритетам *МЗПр*;

регистр запросов прерывания *РЗПр*.

Л Чт/Зп позволяет записать во внутренние регистры управляющие слова «Слово команды инициализации» (СКИ) и «Слово

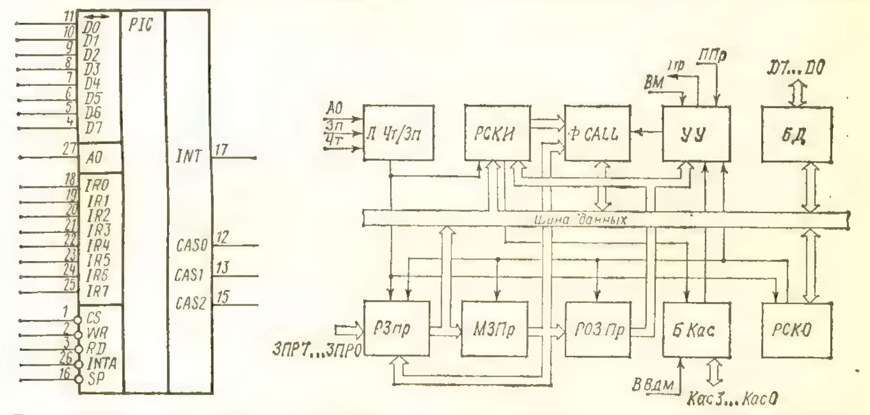


Рис. 3.48. Условное графическое обозначение микросхемы КР580ВН59

Рис. 3.49. Структурная схема микросхемы КР580ВН59

команды операции» (СКО). Сигнал «0» на входе *Чт* позволяет считать содержимое внутренних регистров либо код запроса прерывания на шину данных.

РСКИ предназначен для записи и хранения слов команд инициализации СКИ1, СКИ2.

Если несколько БИС ПКП соединены каскадно, то записываются команды СКИ1, СКИ2, СКИ3. Форматы команд СКО и СКИ приведены на рис. 3.50, 3.51, 3.52. После записи команд инициализации БИС ПКП готова к приему запросов.

ФCALL предназначена для формирования и выдачи трехбайтовой команды CALL (см. табл. 3.9). Побайтовая выдача команды CALL происходит при поступлении сигнала *ППр*. При поступлении первого сигнала *ППр* выдается первый байт команды CALL. При поступлении второго и третьего сигналов *ППр* на *ШD* соответственно выдаются второй и третий байты этой команды.

УУ предназначено для формирования и выдачи сигнала *Пр* после поступления хотя бы одного запроса от внешних устройств (см. рис. 3.49). Кроме этого, в зависимости от наличия каскадно-

Таблица 3.25. **Назначение выводов микросхемы КР580ВН59**

Номер вывода	Назначение	Разряд	Обозначение	Тип
1	Выбор микросхемы	—	ВМ	Вход
2	Запись	—	Зп	Вход
3	Чтение	—	Чт	Вход
4	Шина данных	7	*D*7	Вход-выход
5	Шина данных	6	*D*6	Вход-выход
6	Шина данных	5	*D*5	Вход-выход
7	Шина данных	4	*D*4	Вход-выход
8	Шина данных	3	*D*3	Вход-выход
9	Шина данных	2	*D*2	Вход-выход
10	Шина данных	1	*D*1	Вход-выход
11	Шина данных	0	*D*0	Вход-выход
12	Канал каскадирования 0	—	Кас0	Вход-выход
13	» » 1	—	Кас1	Вход-выход
14	Общий	—	*GND*	—
15	Канал каскадирования 2	—	Кас2	Вход-выход
16	Выбор ведомого	—	ВВдм	Вход
17	Прерывание	—	Пр	Выход
18	Запрос прерывания	0	ЗПр0	Вход
19	Запрос прерывания	1	ЗПр1	Вход
20	Запрос прерывания	2	ЗПр2	Вход
21	Запрос прерывания	3	ЗПр3	Вход
22	Запрос прерывания	4	ЗПр4	Вход
23	Запрос прерывания	5	ЗПр5	Вход
24	Запрос прерывания	6	ЗПр6	Вход
25	Запрос прерывания	7	ЗПр7	Вход
26	Подтверждение прерывания	—	ППр	Вход
27	Адрес	0	А0	Вход
28	Питание	—	U_{CC}	—

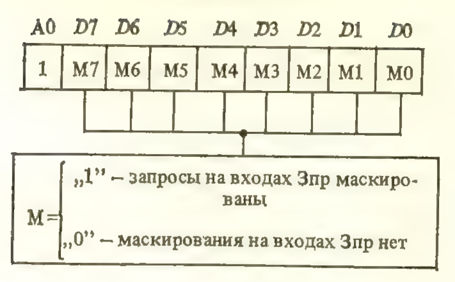
Примечание. Шина данных трехстабильная.

соединенных БИС ПКП процесс выдачи команды CALL различен. При отсутствии каскадного соединения БИС три байта команды вырабатывает одна схема. При наличии каскадного соединения первый байт команды выдает ведущая БИС, второй и третий байты формирует та БИС, запрос которой вызвал сигнал прерывания *ППр*.

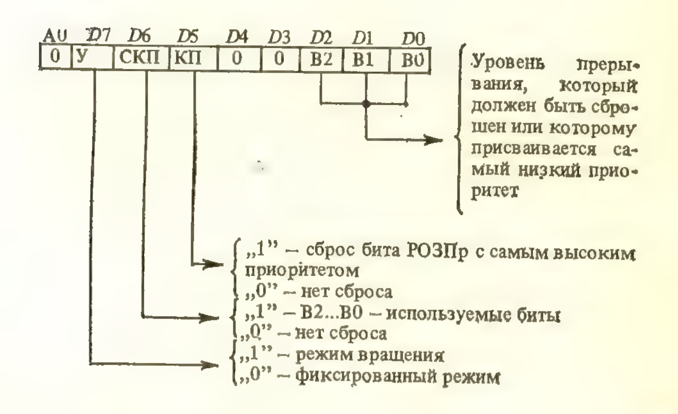
БД предназначен для подключения БИС ПКП к системной шине данных. *ШД* используется для обмена данными, записи управляющих слов СКИ и СКО, чтения (выдачи) содержимого внутренних регистров, чтения двоично-десятичного кода обслуживаемого запроса, чтения байтов команды CALL.

РСКО предназначен для программирования операций БИС ПКП. Запись в *РСКО* осуществляется при наличии сигнала «0» на входах ВМ и Зп. В этом случае в указанный регистр записываются слова команд операций СКО3 ... СКО0, которые задают: режимы обслуживания по запросу и по результатам опроса.

Формат СКО1



Формат СКО2



Формат СКО3

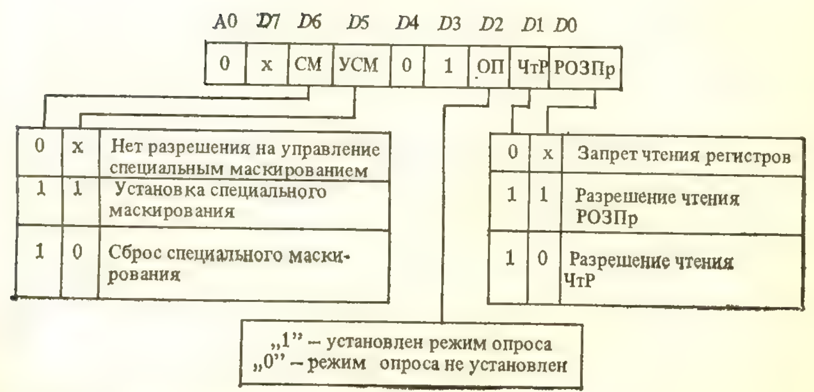
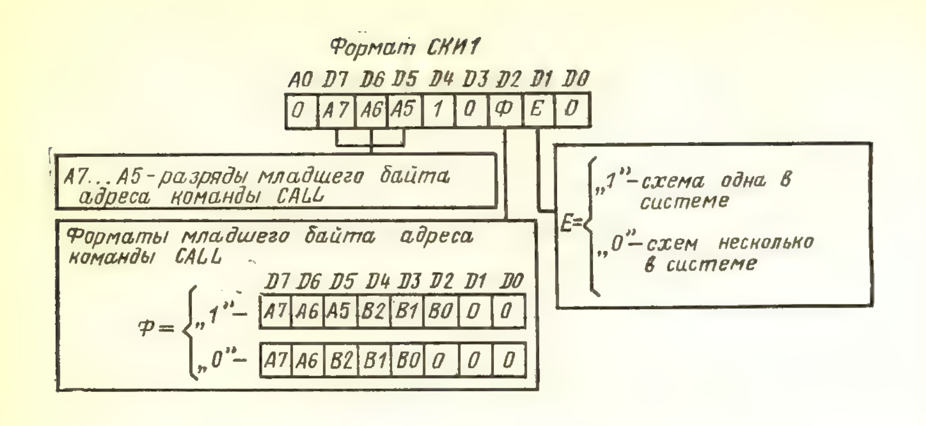


Рис. 3.50. Форматы команд СКО



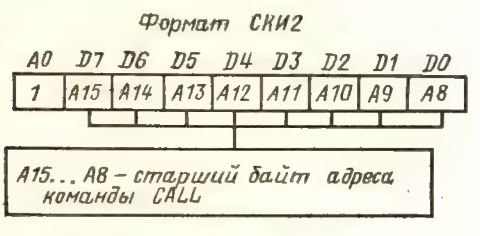


Рис. 3.51. Форматы команд СКИ1, СКИ2

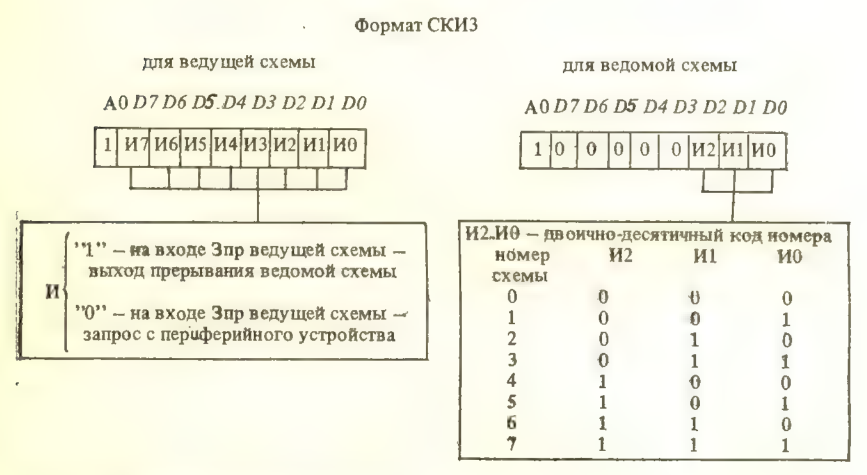


Рис. 3.52. Формат команды СКИ3

Режим обслуживания по запросу предусматривает выполнение следующих операций: маскирование запросов прерывания, фиксирование приоритета, обычный конец прерываний, циклический сдвиг приоритета при обычном конце прерываний, специальный конец прерываний, циклический сдвиг приоритета при специальном конце прерываний, циклический сдвиг приоритета без завершения прерываний, специальное маскирование, чтение *РЗПр*, чтение *РОЗПр*.

Второй режим работы предусматривает выполнение операций: маскирования запросов прерывания, специального маскирования, чтения *РЗПр*, чтения *РОЗПр*.

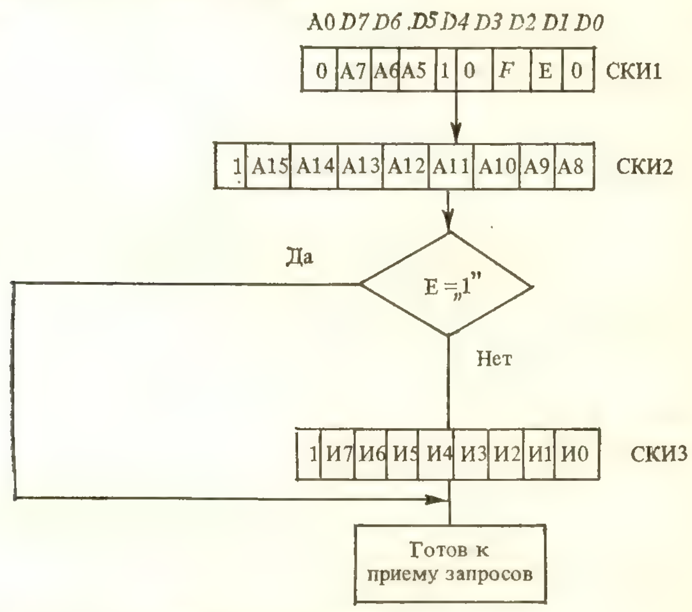


Рис. 3.53. Последовательность команд СКИ

БКас предназначен для обеспечения работы БИС ПКП в каскадном включении. В этом случае одна БИС является ведущей (на входе *ВВдм* сигнал «1»), остальные БИС ПКП — ведомые.

Каждой микросхеме программным путем присваивается номер в каскадном соединении. Двоично-десятичный код номера ведомой микросхемы выдается ведущей по шинам *Кас*2 ... *Кас*0 по заднему фронту первого сигнала *БВдм* и хранится до появления заднего фронта третьего сигнала *ППр*.

БИС ПКП может реализовать различные алгоритмы обслуживания. Система команд БИС ПКП приведена в табл. 3.26.

Перед началом работы БИС ПКП устанавливают в исходное состояние двумя или тремя последовательными командами СКИ1, СКИ2 и СКИ3 (рис. 3.53).

Таблица 3.26. **Система команд микросхемы КР580ВН59**

Команда	Модификация	Разряды кода команды								
		А0	7	6	5	4	3	2	1	0
СКИ1	а	0	А7	А6	А5	1	—	1	1	—
	б	0	А7	А6	А5	1	—	1	0	—
	в	0	А7	—	—	1	—	0	1	—
	г	0	А7	—	—	1	—	0	0	—
СКИ2	—	1	А15	А14	А13	А12	А11	А10	А9	А8
СКИ3	а	1	И7	И6	И5	И4	И3	И2	И1	И0
	б	1	—	—	—	—	—	И2	И1	И0
СКО1	—	1	М7	М6	М5	М4	М3	М2	М1	М0
СКО2	а	0	0	0	1	0	0	—	—	—
	б	0	0	1	1	0	0	*В*2	*В*1	*В*0
	в	0	1	0	0	0	0	—	—	—
	г	0	1	1	0	0	0	*В*2	*В*1	*В*0
СКО3	а	0	—	0	0	0	1	1	0	1
	б	0	—	0	0	0	1	0	1	1
	в	0	—	0	0	0	1	0	1	0
	г	0	—	1	0	0	1	0	0	0

По команде СКИ1 сбрасываются *РЗПр*, *РОЗПр* и *РСКО* (см. рис. 3.49, 3.50). Устанавливается фиксированный статус приоритета. Признаком команды СКИ1 является наличие в *D*4 сигнала «1» и в А0 — «0». В зависимости от признаков «Е» и «Ф» СКИ1 имеет четыре модификации (см. рис. 3.51). При отсутствии каскадирования «Е» установлен в «1», в противном случае — в «0».

Если «Ф» = «1» (*D*2 = «1»), то формат равен 4. Это означает, что адреса программ обслуживания запросов отстоят на 4 адреса. Если «Ф» = «0», то формат равен 8.

При формате 4 в разрядах *D*7 ... *D*5 команды СКИ1 указаны разряды начального адреса подпрограммы обслуживания запросов. При формате 8 в разрядах *D*7 и *D*6 указаны разряды начального адреса обслуживания запросов.

РОЗПр предназначен для хранения сигналов, поступающих со схемы МЗПр (см. рис. 3.49). Разряды *РОЗПр*, соответствующие запросам прерывания, обслуживаемым в данный момент, устанавливаются в состояние «1» при поступлении второго сигнала *ППр*. Эти разряды РОЗПр сохраняют свое состояние до выдачи команды DI (см. табл. 3.9) центральным процессором.

РЗпр предназначен для записи и хранения запросов прерывания от внешних устройств. Запись в соответствующий разряд регистра происходит при изменении сигнала на входе из «0» в «1». Сигнал «1» должен удерживаться до появления первого сигнала *ППр*. Установленный разряд, соответствующий обслуживаемому запросу, возвращается в исходное («0») состояние по второму сигналу *ППр*.

Схема *МЗПр* разрешает (запрещает) прохождение сигналов запросов с выходов *РЗПр* на *РОЗПр*. Схема может маскировать принятые запросы. Код маски записывается по команде СКО1 и хранится так до записи нового кода в *РСКО* (см. табл. 3.26).

После команды СКИ1 следует СКИ2. В ней указываются старшие разряды (15 ... 8) 16-разрядного адреса программы обслуживания.

Если разряд «Е» = «0» и А0 = «1», то команда, следующая за СКИ2, воспринимается БИС ПКП как СКИ3. Эта команда имеет две модификации — СКИ3а и СКИ3б.

Команда СКИ3а подается на ведущую БИС; СКИ3б — на ведомую.

Если в разрядах *D*7 ... *D*0 команды СКИ3а записан «0», это означает, что на соответствующий вход подается запрос от периферийного устройства. В разрядах *D*2 ... *D*0 команды СКИ3б указан идентификатор ведомой микросхемы, который должен быть равен номеру ввода ведущей микросхемы, к которому подключена ведомая. Так, например, если выход *Пр* ведомой микросхемы подключен ко входу *ЗПр6*, то в разрядах *D*2 ... *D*0 команды СКИ3б, выдаваемых на эту БИС, указан двоичный код 110.

Таким образом, перед обслуживанием запросов в БИС ПКП должна быть выдана последовательность команд СКИ (см. рис. 3.53).

Выбор или изменение в процессе работы алгоритма обслуживания запросов осуществляется с помощью команд СКО.

Команда СКО1 (признак А0 = «1») позволяет загружать регистр маски по сигналу *Зп* = «0» (см. рис. 3.50).

Установка признака M_i = «1» в этой команде указывает на блокировку обслуживания соответствующего запроса.

Содержимое регистра маски выдается на шину данных при подаче на входы микросхемы следующих сигналов: ВМ = «0» Чт = «0», А0 = «1».

Команда СКО2 (признаки А0 = «0», *D*4 = «0», *D*3 = «0») имеет модификации — СКО2а ... СКО2г (табл. 3.26). Группа команд СКО2 указывает вид конца обслуживания прерывания, а также вид установки для приоритета кольца (табл. 3.27).

Бит «*У*» (*D*7) команды СКО2 определяет тип установки приоритета: фиксированный (*У* = «0») или вращающийся (*У* = «1»).

Имеются два варианта режима с вращающимися приоритетами (рис. 3.50):

Таблица 3.27. Варианты статусов уровней приоритета

Обозначение входов микросхемы	Статусы уровней приоритета							
Зпр0	7 *	6	5	4	3	2	1	0
Зпр1	0	7	6	5	4	3	2	1
Зпр2	1	0	7	6	5	4	3	2
Зпр3	2	1	0	7	6	5	4	3
Зпр4	3	2	1	0	7	6	5	4
Зпр5	4	3	2	1	0	7	6	5
Зпр6	5	4	3	2	1	0	7	6
Зпр7	6	5	4	3	2	1	0	7

* «Дно» приоритетного кольца.

1. Устройству, получившему обслуживание, назначается наинизший приоритет. Разряд РОЗПр соответствующего запроса сбрасывается в нуль, в то время как остальные приоритеты сдвигаются по кольцу. В результате устройство, запрашивающее прерывание, должно ожидать время, необходимое для обслуживания не более семи других устройств по одному разу.

2. Статус уровня приоритета задается программно, т. е. фиксируется фактический уровень приоритета («дно» кольца вращения).

Установленное значение статуса приоритета сохраняется до подачи команды, которая может его изменить. Биты КП (конец прерывания) и СКП (специальный конец прерывания) в командах СКО2 определяют характер воздействия команды на *РОЗПр*. Бит КП = «1» сбрасывает в *РОЗПр* бит с наибольшим приоритетом среди установленных, который для фиксированного приоритета соответствует последнему обслуживаемому запросу (рис. 3.50 ... 3.53).

В случае вращающегося приоритета используется бит СКП. Наличие «1» в нем указывает на необходимость сбросить в *РОЗПр* бит, соответствующий номеру запроса, указанному в поле *В2* ... *В0* команды СКО2. Команда СКО2а применяется в случае фиксированного уровня приоритета при обычном конце прерывания.

Команда СКО2б применяется в случае фиксированного уровня прерывания при специальном прерывании. Команда сбрасывает в состояние «0» тот разряд РОЗПр, который указан в разрядах *D*2 ... *D*0 этой команды.

Команда СКО2в (обычный конец прерывания) вводит установку статуса приоритета по последнему обслуживаемому запросу. Разряд *РОЗПр*, соответствующий этому запросу, сбрасывается в нуль, и ему присваивается наинизший уровень приоритета.

Команда СКО2г задает статус приоритета без выполнения операции конца прерывания. Код в разрядах *D*2 ... *D*0 указывает «дно» приоритетного кольца.

Группа команд типа СКО3 (A0 = «0», $D5$ = «0», $D4$ = «0») используется в режиме чтения и установки специального маскирования (см. рис. 3.50). Команда СКО3 фактически имеет четыре модификации СКО3а ... СКО3г (см. табл. 3.26).

Команда СКО3а устанавливает режим обслуживания по результатам опроса. После подачи на вход *Чт* микросхемы сигнала «0» действие команды СКО3а прекращается, а действия команд СКО3б и СКО3в сохраняются.

Команда СКО3г используется совместно с командой СКО1 в том случае, если необходимо обслужить запрос. Запрос блокирован старшим по уровню приоритета обслуженным или обслуживаемым запросом, хранимым в *РОЗПр*, без сброса последнего (специальное маскирование).

Приведенная система команд позволяет устанавливать различные алгоритмы и закреплять приоритет за внешними устройствами как статически, так и динамически, т. е. в любое время работы основной программы.

Прерывание по запросам возможно после установки в исходное состояние БИС ПКП.

Все запросы на прерывание, поступающие от внешних устройств, подаются на входы запросов БИС ПКП и записываются в соответствующие разряды *РЗПр*. Наличие запросов воспринимается микросхемой при переходе входных сигналов запросов от «0» к «1», причем сигнал «1» должен сохраниться по крайней мере до прихода первого сигнала *ППр*.

Регистр *РЗПр* через схему *МЗПр* соединяется с регистром *РОЗПр*.

Схема *МЗПр* анализирует поступающие запросы по уровню приоритетов и выдает разрешение микросхемы на выдачу сигнала *Пр*.

Запрос с наивысшим уровнем приоритета, прошедший через схему *МЗПр*, записывается в *РОЗПр*. Запросы, поступающие с *РЗПр*, могут быть маскированы в регистре маски командой СКО1, устанавливающей в «1» маскируемые разряды. Замаскированные запросы не поступают на схему анализа по уровню приоритета и, следовательно, не влияют на запросы прерываний более низкого уровня.

Как указывалось, регистр *РЗПр* используется для хранения всех запросов на прерывание, регистр *РОЗПр* — для хранения всех тех запросов на прерывание, которые обслуживались или обслуживаются в данный момент. Если запрос, поступивший на микросхему, получил подтверждение (сформировался сигнал *Пр*), то он блокирует запросы с более низкими уровнями приоритетов даже в том случае, если указанный запрос после получения сигнала *ППр* был замаскирован. Для того чтобы запросы с более низкими уровнями приоритетов получили возможность обслуживания, можно либо подать на микросхему одну из команд СКО2 и установить в состояние «0» соответствующий бит *РОЗПр*, либо

подать на микросхему команду установки специального маскирования (СКОЗг).

Как отмечалось, последовательность сигналов *ППр*, которые выдаются на микросхему в ответ на сигнал *Пр*, вызывает выдачу на шину данных трехбайтной команды CALL, в которой указывается адрес подпрограммы обслуживания запросам.

Младший байт адреса содержит три части (см. рис. 3.51):

разряды *A7*, *A6* или *A7*, *A6*, *A5* в зависимости от запрограммированного формата заранее записываются в БИС ПКП;

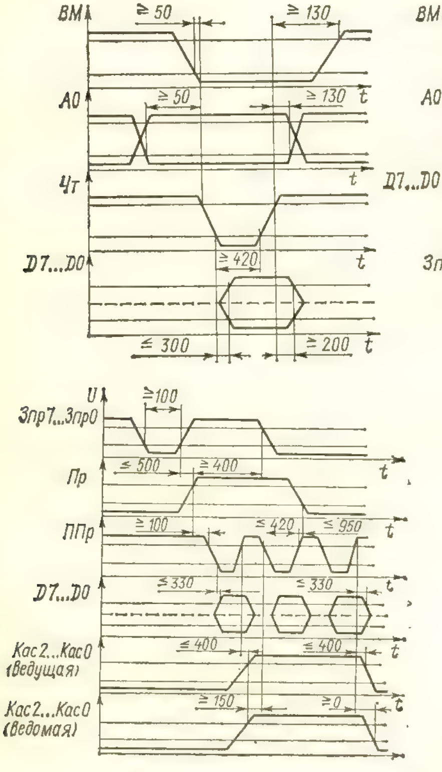


Рис. 3.54. Временна̀я диаграмма работы микросхемы КР580ВН59 при чтении и записи

Рис. 3.55. Временна̀я диаграмма работы микросхемы КР580ВН59 при обслуживании запроса

разряды *A5*, *A4*, *A3* или *A4*, *A3*, *A2* (в зависимости от запрограммированного формата) являются автоматически выдаваемыми номерами входа запроса, получившего подтверждение;

разряды *A2*, *A1*, *A0* или *A1*, *A0* (в зависимости от запрограммированного формата) устанавливаются в «0».

Код старшего байта, записанный в микросхему, заранее выдается из регистра *РСКИ* (СКИ2).

В системах, выполненных на базе МП серии КР580, по команде CALL основная программа прерывается и начинается выполнение подпрограммы обслуживания данного запроса.

Таблица 3.28. Статические параметры микросхемы КР580ВН59

Параметр, единица измерения	Обозначение	Значения		Режим измерения
		мин.	макс.	
Входное напряжение «1», В	U_{IH}	2,4	—	—
Выходное напряжение «0», В	U_{OL}	—	0,45	—
Выходное напряжение «1», В	U_{OH}	2,4	—	—
Входной ток на выводах *Зпр*7 ... *Зпр*0, мкА	I_{I1}	—300	—50	$U_{IL} = 0,4$ В
Ток утечки на входах, мкА	I_{IZ}	—	50	—
Выходной ток в состоянии «Выключено», мкА	I_{OZ}	—	±50	—
Ток потребления от источника питания, мА	I_{cc}	—	100	—
Входная емкость, пФ	C_I	—	10	$f = 1$ МГц
Емкость входа-выхода, пФ	C_{IO}	—	20	—

После выполнения подпрограммы обслуживания запроса необходимо произвести возврат к прерванной программе.

Прерывание по результатам опроса осуществляется по инициативе программы микропроцессорной системы путем подачи на БИС ПКП команды СКОЗа и последующей подачи на вход *Чт* сигнала «0» (см. табл. 3.26, рис. 3.52). Сигнал *Пр* = «1» свидетельствует о наличии прерывания.

После обслуживания данного запроса на БИС ПКП обычно выдается одна из команд СКО2, которая устанавливает соответствующие разряды регистра *РОЗПр* в состояние «0».

При этом СКО2 устанавливает вид задания статуса приоритета и «дно» приоритетного кольца. Прерывание по результатам опроса используется в случае существования одной подпрограммы обслуживания всех запросов.

Временные диаграммы работы БИС ПКП приведены на рис. 3.54 и 3.55.

В табл. 3.28 приведены статические параметры микросхемы.

3.8. ПРИМЕРЫ ПРИМЕНЕНИЯ МИКРОСХЕМ СЕРИИ КР580

Типовая микропроцессорная система, построенная на основе микросхем серии КР580, в общем случае состоит из модуля центрального процессора *ЦП*, постоянного запоминающего устройства *ПЗУ* или *ППЗУ*, оперативного запоминающего устройства ОЗУ, устройства ввода-вывода *УВВ*. Структурная схема такой системы показана на рис. 3.56.

Модуль *ЦП* производит необходимые действия по программе, управляет передачей данных в систему и выдачей данных из нее, вырабатывает необходимый набор управляющих сигналов для всех устройств системы.

Связь между устройствами системы осуществляется по шинам адреса и данных.

Все управляющие сигналы системы объединены в шину управляющих сигналов.

Шина данных — двунаправленная, предназначена для передачи информации между модулями *ЦП* и *ЗУ* или *УВВ*.

Шина адреса — однонаправленная. Имеющаяся на ней информация указывает определенную область *ЗУ* или *УВВ*, к которым обращается модуль *ЦП*.

Шина управляющих сигналов — однонаправленная, содержит набор сигналов, которые вырабатывает модуль *ЦП* и сигналы, которые вырабатывают внешние устройства.

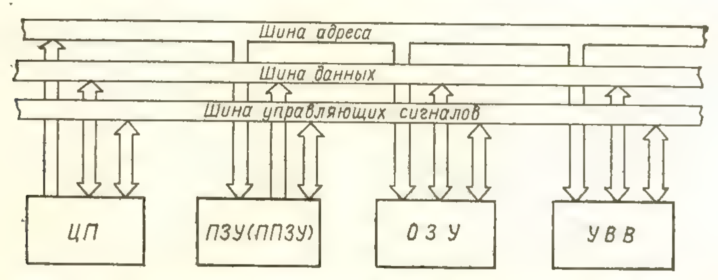


Рис. 3.56. Обобщенная структура системы на базе микросхемы КР580ИК80

Модуль *ЦП* в простейшем случае состоит из центрального обрабатывающего устройства и генератора тактовых импульсов.

Центральный процессор устанавливается в исходное (нулевое) состояние по программе или при включении питания. Начиная от исходного состояния *ЦП* работает по программе до тех пор, пока на вход *Готовность* МП подается «1» и не поступает сигнал прерывания или сигнал *Захват*, выдаваемые внешними устройствами.

При работе с *ЗУ* или *УВВ ЦП* выполняет операции в такой последовательности: выдает управляющие сигналы на шину управляющих сигналов и двоичный код на шину адреса, определяет области *ЗУ* или *УВВ*, которые будут использованы в данном цикле обращения, получает информацию из выбранного *ЗУ* или *УВВ* или передает им информацию, производит обработку информации, передает (если необходимо по программе) информацию в *ЗУ* или *УВВ*.

Центральный процессор получает из *ПЗУ* команды программы и производит передачу данных в *ОЗУ* и *УВВ* или получает данные из них.

Количество команд в программе может быть различным. Они определяют тип операций, которые должны быть выполнены в микропроцессоре, а также направление передачи информации между модулем *ЦП* и другими устройствами системы.

Большое влияние на быстродействие системы оказывают время выборки информации из *ЗУ* и время обработки информации в МП.

Независимо от скорости обработки информации в *ЦП* эффективность работы *ЦП* будет значительно снижена, если для работы с УВВ требуется выполнение сложных и длительных подпрограмм. Именно поэтому простота и экономичность связи *ЦП* с другими элементами системы становится одним из основных факторов, который необходимо учитывать при построении микропроцессорных систем.

Одна из важных особенностей системы, построенной на базе микропроцессорного комплекта БИС серии КР580 — гибкость и мощность ее структуры ввода-вывода.

Основные команды, по которым происходит обращение к *УВВ* — IN и OUT. Эти команды разделяют области адресов *ЗУ* и *УВВ* и с их помощью формируются в схеме выработки управляющих сигналов сигналы обращения к *УВВ*.

Возможна и другая организация структуры ввода-вывода, при которой УВВ рассматривается как часть массива *ЗУ*, т. е. имеет обобщенное поле адресов *ЗУ* и *УВВ*. Рассматривая область адресов УВВ как область адресов *ЗУ*, можно использовать для обращения к УВВ такие же команды, как и при обращении к *ЗУ*. При этом новые управляющие сигналы для УВВ можно получить, объединяя имеющиеся сигналы (*ЧтП*, *ЗпП*) со старшим разрядом адреса А15.

При работе микропроцессорной системы под стековую память отводится часть массива адресов *ОЗУ*. Можно выделить под стековую память отдельное ЗУ, что расширит объем основного *ОЗУ* и облегчит программирование системы. Сигналы обращения к этому *ЗУ* можно формировать, объединяя сигнал внутреннего состояния БИС КР580ИК80А *Стек* с сигналами *Чтение памяти* и *Запись в память*.

Значительно расширить объем используемого ЗУ можно методом страничной организации *ЗУ*.

В пределах каждой страницы можно использовать возможность обращения БИС *ЦП* к *ЗУ* емкостью до 64 К слов. В этом случае необходимы внешний регистр с дешифратором для выбора той или иной страницы памяти.

Связь между устройствами системы имеет магистральную структуру. Если система не является магистрально-организованной, то это может потребовать использования большого количества внешней аппаратуры для распределения информации по параллельным входам и выходам. Применение магистральной организации уменьшает число необходимых схем сопряжения, обеспечивает возможность расширения системы и позволяет реализовать режим прямого доступа к памяти.

8-разрядная шина данных состоит из восьми двунаправленных линий, т. е. по одним и тем же линиям информация может как

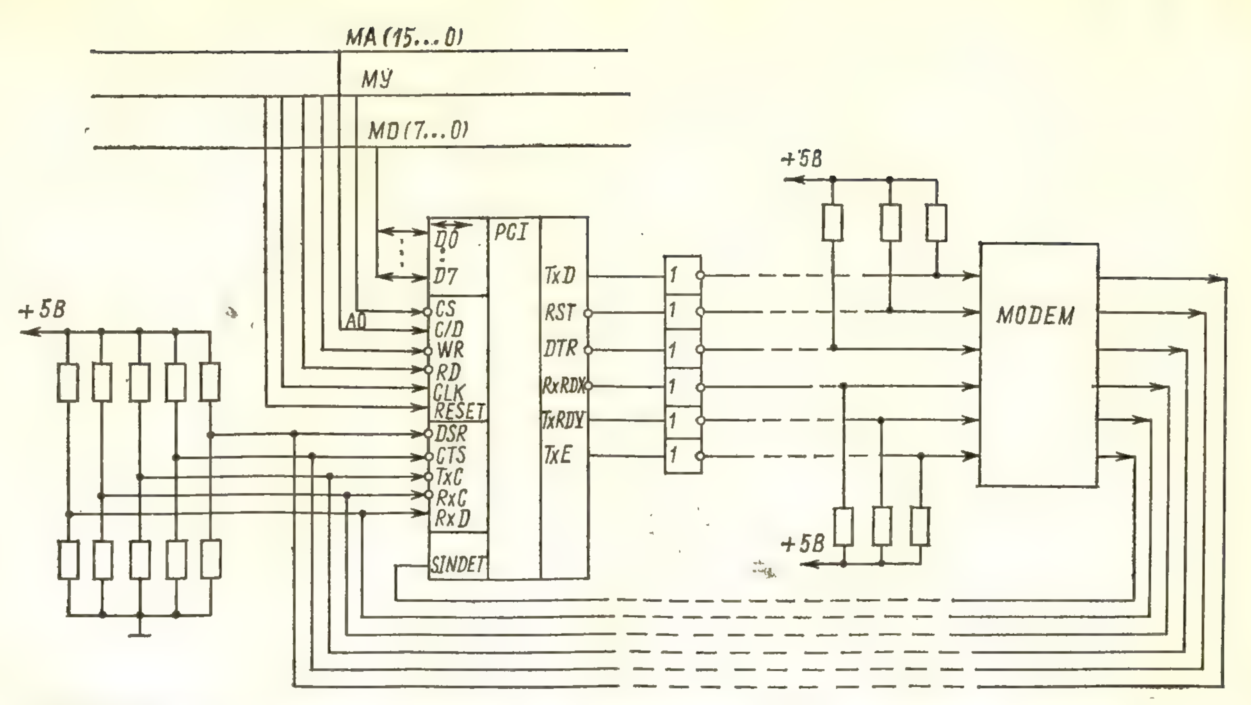


Рис. 3.57. Пример организации совместной работы микросхемы КР580ИК51 и внешнего устройства

приниматься, так и передаваться относительно одного и того же устройства.

Магистраль обеспечивает три типа обмена информацией: программный в режиме прямого доступа к памяти (ПДП) и в режиме прерывания программы.

Программный обмен информацией — это передача информации по инициативе и под управлением программы.

Обмен информацией по инициативе внешних устройств может выполняться в режиме прямого доступа к памяти и в режиме прерывания программы. Обмен в режиме прямого доступа к па-

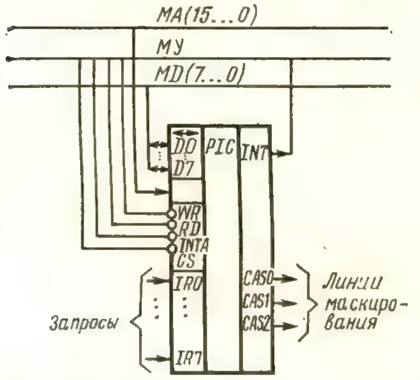


Рис. 3.58. Пример подключения микросхемы КР580ВН59

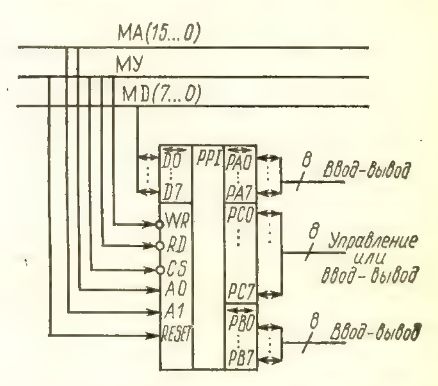


Рис. 3.59. Пример подключения микросхемы КР580ИК55 к внешним устройствам и шинам системы

мяти является самым быстрым способом передачи данных между ЗУ и внешними устройствами. При этом обмене не изменяется внутреннее состояние БИС МП. Поэтому обмен данных может выполняться в интервале времени, когда БИС МП под действием сигнала *Захват*, выдаваемого внешними устройствами, передает управление шинами внешним устройствам. Адресация и управление размерами передаваемого массива данных находится под управлением устройства, получившего прямой доступ к памяти без вмешательства МП.

Устройство, запросившее прямой доступ к *ЗУ*, само вырабатывает сигналы управления. Массивы данных в режиме прямого доступа к *ЗУ* могут передаваться со скоростью, определяемой быстродействием *ЗУ*.

Обмен информацией в режиме прерывания программы — это выполнение программы обслуживания по требованию внешнего устройства. Центральный процессор при этом приостанавливает выполнение текущей программы, чтобы обслужить запрашивающее устройство. После завершения выполнения программы обслуживания прерывания МП возобновляет выполнение основной программы с того места, где она была прервана.

Каждое устройство микропроцессорной системы имеет приемники и передатчики информации. В модуле *ЦП* передатчиками являются буферы адреса и данных. Приемниками являются буферы данных.

Выдачу информации на магистраль в любой интервал времени может производить только один передатчик. Прием информации может осуществляться одним или несколькими приемниками, поэтому невозможна параллельная работа нескольких пар схем передатчик—приемник. Это накладывает ограничение на быстродействие системы.

Наиболее целесообразным при использовании БИС серии КР580 является способ организации шины с применением передатчиков с трехстабильным выходом. Использование третьего состояния позволяет просто принимать информацию с шины или выдавать ее на шину, переводя в отключенное состояние буферы адреса и данных тех устройств системы, которые не участвуют в данный момент в обмене информацией.

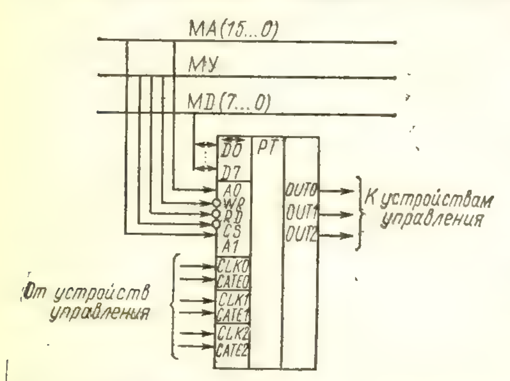


Рис. 3.60. Пример включения микросхемы КР580ВИ53

Значительно упрощается организация шин при применении в системе в качестве буферов адреса и данных шинных формирователей К589АП16 или К589АП26.

Шинные формирователи можно также применять в качестве буферов адреса микропроцессора, настраивая их с помощью управляющих сигналов только для выдачи адреса на магистраль. На рис. 3.57—3.60 даны примеры подключения БИС серии КР580 к системным шинам и внешним устройствам.

На рис. 3.59, 3.60 магистрали адреса, данных и управления обозначены соответственно *МА*, *MD* и *МУ*.

Глава 4

МИКРОПРОЦЕССОРНЫЙ КОМПЛЕКТ БИС СЕРИИ К1801

4.1. ОБЩИЕ СВЕДЕНИЯ

МПК БИС серии К1801 предназначен для построения разнообразных систем [23, 24, 50, 52, 55, 59]. Комплект разработан на базе *n*-МОП-технологии, включает микросхемы 6 типов с напря-

Таблица 4.1. Состав МПК БИС серии К1801

Тип микросхемы	Тип корпуса	Функциональное назначение
К1801ВМ1	429.42-5	Микропроцессор
К1801ВП1-030	429.42-5	Устройство управления ОЗУ динамического типа
К1801ВП1-033	429.42-5	Многофункциональное устройство, реализующее интерфейс накопителя на гибких магнитных дисках, контроллер интерфейса параллельного ввода-вывода, контроллер байтового параллельного интерфейса ввода-вывода ИРПР.
К1801ВП1-034	429.42-5	Многофункциональное устройство, реализующее устройство передачи информации, устройство выдачи вектора прерывания и компаратор адреса, буферный регистр
К1801ВП1-035	429.42-5	Контроллер внешних устройств, работающих на линии связи с последовательной передачей информации
К1801РЕ1-000	2106.24-1	Системное ПЗУ

жением питания +5 В (±5 %). Основой комплекта являются БИС микропроцессора К1801ВМ1 с системой команд микроЭВМ «Электроника-60» и матричная БИС К1801ВП1. Матричная БИС представляет собой базовый кристалл, содержащий матрицу стандартных логических элементов. Элементы матрицы в зависимости от поставленной задачи могут быть соединены между собой необходимым образом, причем осуществляют это при изготовлении БИС коммутацией верхних слоев металлизации. К схемам на базе матричной БИС относятся: БИС К1801ВП1-030, К1801ВП1-033, К1801ВП1-034, К1801ВП1-035.

Матричная БИС позволяет значительно сократить время, затрачиваемое на разработки специализированных устройств.

Таблица 4.2. Предельно допустимые режимы работы БИС серии К1801

Параметр	Обозначение	Тип микросхемы		
		К1801ВМ1	К1801ВП1	К1801РЕ1
Напряжение питания, В	U_{CC}	$4{,}5 \leqslant U_{CC} \leqslant 5{,}5$	$U_{CC} \leqslant 7$	$U_{CC} \leqslant 7$
Напряжение входного сигнала, В	U_I	$0 \leqslant U_I \leqslant U_{CC}$	$0 \leqslant U_I \leqslant U_{CC}$	$0{,}5 \leqslant U_I \leqslant 7$
Емкость нагрузки, пФ	C_L	$C_L \leqslant 100$	$C_L \leqslant 100$	—
Электростатический потенциал, В	φ	$\varphi \leqslant 100$	$\varphi \leqslant 100$	$\varphi \leqslant 100$

Состав МПК БИС серии К1801 дан в табл. 4.1. Предельно допустимые режимы эксплуатации микросхем серии К1801 приведены в табл. 4.2.

МПК БИС серии К1801 может быть использован в программируемых «интеллектуальных» контроллерах судовых систем управления навигационными приборами, узлами аппаратуры связи и локации. МПК можно успешно применять в многопроцессорных комплексах с системой команд типа «Электроника-60».

4.2. МИКРОСХЕМА К1801ВМ1

Представляет 16-разрядный микропроцессор, предназначенный для обработки цифровой информации.

МП имеет: 16-разрядную совмещенную шину адреса и данных; 8 внутренних регистров общего назначения; возможность адресации к памяти объемом 64 Кбайт; 4 уровня запросов прерывания.

МП выполняет 69 команд, из них 67 входит в систему команд микроЭВМ «Электроника-60».

Структура команд — безадресная, одноадресная и двухадресная. При обращении к ячейкам памяти (регистрам) используют 4 основных метода адресации: регистровый; автоинкрементный; автодекрементный; индексный.

Эти методы включают два типа адресации: прямую и косвенную. МП может взаимодействовать с тремя такими же процессорами в мультипроцессорной структуре. Условное графическое обозначение микросхемы приведено на рис. 4.1, структурная схема — на рис. 4.2, назначение выводов — в табл. 4.3.

Рис. 4.1. Условное графическое обозначение микросхемы К1801ВМ1

В ее состав входят следующие блоки: операционный *ОБ*; микропрограммного управления *БМУ*, прерываний *БПР*; интерфейсный *ИБ*; согласования *БС*, буферный регистр команд *БК*.

ОБ предназначен для выполнения следующих операций: вычисление и хранение адреса; прием и хранение данных; выполнение межрегистровых арифметико-логических операций; выдача данных на системную шину; формирование состояний. Основу *ОБ* составляют: СОЗУ из 14- и 16-разрядных внутренних регистров (РОН), 16-разрядного АЛУ, регистр состояния процессора (РСП), регистр адреса и данных.

Восемь из 14 регистров СОЗУ операционного блока программно доступны. Этим же свойством обладает РСП. Остальные регистры СОЗУ доступны только микропрограммно.

В АЛУ выполняются арифметические и логические операции и вырабатываются признаки, необходимые для формирования состояния. После выполнения указанных операций происходит сдвиг (если это необходимо) на один разряд.

Связь *ОБ* с внутренней шиной МП осуществляется через регистры адреса и данных.

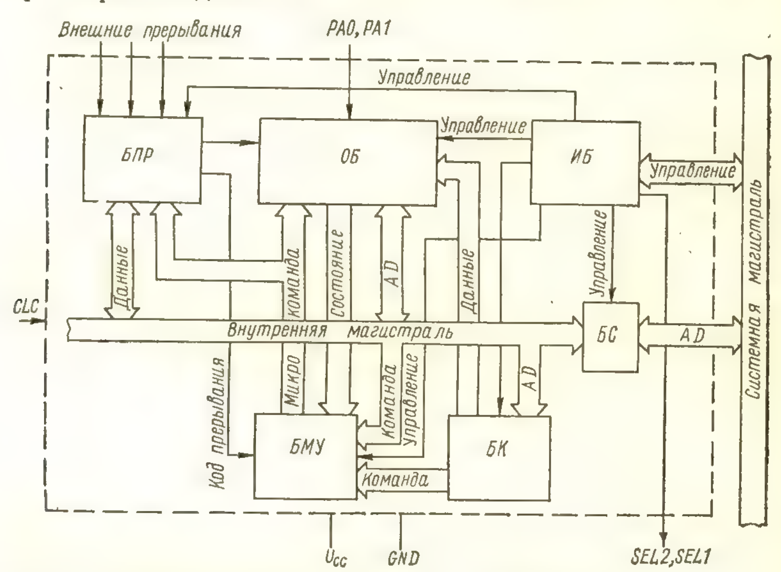


Рис. 4.2. Структурная схема микросхемы К1801ВМ1

БМУ предназначен для преобразования команды, поступившей на МП, в последовательности микрокоманд. Основу *БМУ* составляет программируемая логическая матрица (ПЛМ), в которой «жестко зашиты» микропрограммы команд МП. ПЛМ имеет 250 логических произведений.

БПР предназначен для организации логики приоритетных прерываний в МП. В МП принимаются и проходят первичную обработку внешние и внутренние запросы на прерывание.

ПЛМ, входящая в состав *БПР*, содержит в закодированном виде ответные реакции на запросы прерываний с учетом их приоритетности. Информация с выхода *БПР* поступает на вход *БМУ*. *ИБ* обеспечивает обмен между внешней шиной и *Внутренней магистралью* МП.

В *ИБ* осуществляется анализ положения данного МП в многопроцессорной системе («Главный» или «Не главный»). В зависимости от этого меняются характеристики отдельных выводов. Это относится к сигналам *DMR*, *SACK*, *DMGI*, *DMGO*. Первые два

Таблица 4.3. Назначение выводов микросхемы К1801ВМ1

Номер вывода	Назначение	Разряд	Обозначение	Тип
1	Тактовый импульс	—	*CLC*	Вход
2	Подтверждение запроса магистрали	—	*SACK*	Вход-выход
3	Разрешение захвата	—	*DMGI*	Вход
4		—	*DMGO*	Выход
5	Запрос захвата	—	*DMR*	Вход-выход
6	Резервный контакт	—	*SP*	—
7	Выборка первого регистра ввода-вывода	—	*SEL1*	Выход
8	Выборка второго регистра ввода-вывода	—	*SEL2*	
9		0	*AD0*	
10		1	*AD1*	
11		2	*AD2*	
12		3	*AD3*	
13		4	*AD4*	
14		5	*AD5*	
15	Шина адрес-данные	6	*AD6*	Вход-выход
16		7	*AD7*	
17		8	*AD8*	
18		9	*AD9*	
19		10	*AD10*	
20		11	*AD11*	
21	Общий	—	*GND*	—
22	Шина адрес-данные	12	*AD12*	Вход-выход
23		13	*AD13*	
24		14	*AD14*	
25		15	*AD15*	
26	Адрес процессора	1	*PA1*	Вход
27		0	*PA2*	
28	Шина занята	—	*BSY*	Выход
29	Авария источника питания	—	*DCL0*	Вход
30	Авария сетевого питания	—	*ACL0*	
31	Запрос радиального прерывания	1	*IRQ1*	
32		2	*IRQ2*	
		2	*IRQ2*	
33		3	*IRQ3*	
34	Установка	—	*INIT*	Вход-выход
35	Запрос на векторное прерывание	—	*VIRQ*	Вход
36	Разрешение прерывания	—	*IAKO*	Выход
37	Вывод данных	—	*DOUT*	
38	Ввод данных	—	*DIN*	
39	Ответ приемника	—	*RPLY*	Вход
40	Запись-байт	—	*WTBT*	Выход
41	Синхронизация обмена	—	*SYNC*	Вход-выход
42	Напряжение питания	—	U_{cc}	—

Примечание. Шина адрес-данные, а также выводы 36, 37, 38, 40, 41 — трехстабильные. Выводы 2, 5, 28, 34 имеют «свободный сток».

сигнала в зависимости от того, является ли данная микросхема главной или не главной, будут входами или выходами. Кроме того, *ИБ* управляет *системной магистралью* (обрабатывая и формируя сигналы управления), а также осуществляет арбитраж запросов на обмен по принципу «первым пришел — первым обслужился».

БС управляет приемопередатчиками шины адрес-данные. *БК* служит для приема и хранения текущей команды. В МП осуществляется совмещение процессов обработки текущей и приема следующей команды. Текущая команда хранится в *БМУ*, а следующая может приниматься в *БК*.

4.2.1. Функционирование МП

К особенностям МП относится, в первую очередь, временно́е мультиплексирование подачи адреса и данных. Это означает, что при выполнении операций обращения к каналу, которые присутствуют в каждой команде, сперва происходит выдача адреса, затем по тем же линиям *AD* (см. рис. 4.2) выдаются или принимаются данные. Операции обращения к каналу включают следующие циклы: «Чтение», «Запись», «Чтение — модификация — запись».

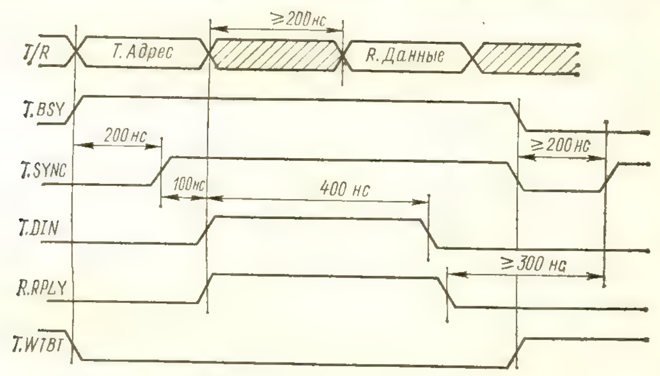


Рис. 4.3. Цикл «Чтение»

При описании временных диаграмм буква *T* перед сигналом означает передатчик, *R* — приемник.

При выполнении цикла «Чтение» (рис. 4.3) данные передаются от пассивного устройства к активному, т. е. к МП. Порядок выполнения этого цикла следующий:

1) МП выставляет адрес и сигнал *BSY*;

2) через 200 нс выставляется сигнал *SYNC* и пассивное устройство запоминает адрес;

3) через 100 нс после сигнала *SYNC* адрес снимается, и МП выставляет сигнал *DIN*;

4) МП настраивается на прием содержимого считываемой ячейки; если за 64 такта синхрочастоты не поступит сигнал *RPLY*, стробирующий считываемые данные, то МП переходит к прерыванию по ошибке обращения к каналу;

5) после поступления от внешнего устройства сигнала *RPLY* МП принимает данные и сбрасывает сигнал *DIN*;

6) после сброса сигнала *DIN* внешнее устройство сбрасывает сигнал *RPLY*;

7) после этого МП снимает сигналы *BSY* и *SYNC*;

8) МП может начать следующий цикл обращения к каналу.

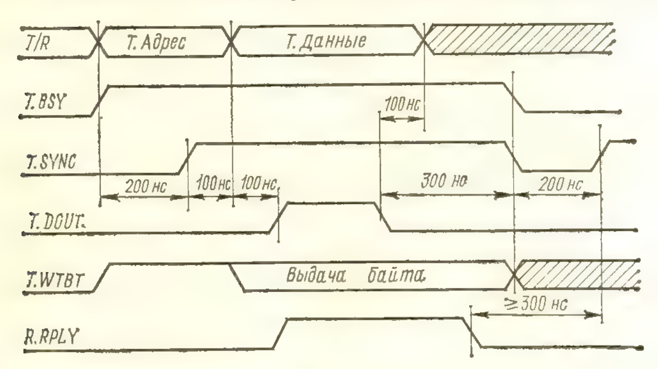


Рис. 4.4. Цикл «Чтение — модификация — запись»

При выполнении цикла «Запись» адрес и данные передаются из МП. В цикле «Запись» происходит следующее:

1) МП выставляет адрес и сигналы *BSY* и *WTBT*;

2) через 200 нс МП выдает сигнал *SYNC*;

3) через 100 нс после сигнала *SYNC* процессор снимает адрес и сигнал *WTBT*; сигнал *WTBT* идентифицирует цикл «Запись»;

4) МП выставляет на шину данные и через 100 нс — сигнал *DOUT*; если после 64 тактов синхрочастоты приемник не выдаст сигнал *RPLY*, то МП переходит к обработке прерывания по ошибке обмена с каналом; при выдаче байта данных активизируется сигнал *WTBT*;

5) через 300 нс после появления сигнала *RPLY* МП снимает сигнал *DOUT*; через 100 нс МП снимает данные;

6) пассивное устройство (приемник) снимает сигнал *RPLY*;

7) МП снимает сигналы *SYNC* и *BSY* и может начать новый цикл обращения к каналу.

Первая часть цикла «Чтение — модификация — запись» аналогична циклу «Чтение». После его окончания сигналы *SYNC* и *BSY* (рис. 4.4) не снимаются, что позволяет произвести запись данных во второй части цикла. Запись начинается после снятия пассивным устройством сигнала *RPLY*. В случае отсутствия этого сигнала действие МП аналогично его действию в циклах «Чтение» и «Запись».

Как указывалось, микросхема К1801ВМ1 может работать в многопроцессорной системе в качестве главного процессора, либо подчиненного (не главного). В этих условиях особое значение имеет алгоритм захвата шины адрес-данные. Рассмотрим, как происходит захват шины главным и не главным устройствами (рис. 4.5).

Не главное устройство сигналом *DMR* запрашивает разрешение на захват шины у главного, который формирует сигнал разрешения *DMGO*. Этот сигнал проходит по цепи последовательно через каждый не главный МП. В случае, если сигнал *DMR* выставила не данная микросхема, то происходит транслирование

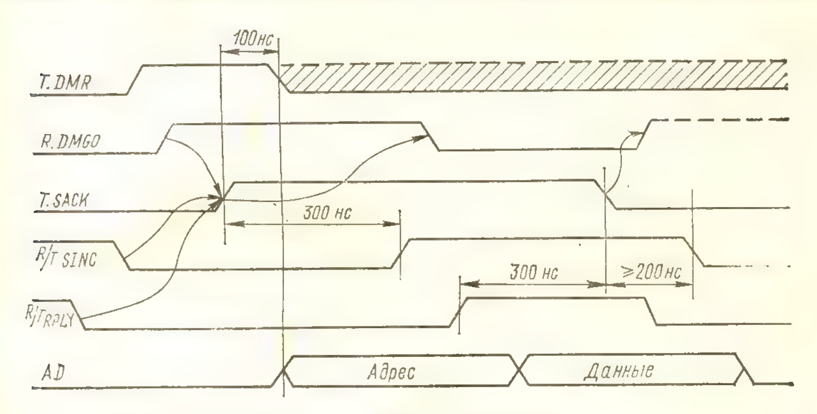


Рис. 4.5. Выполнение процедуры захвата магистрали

сигнала разрешения на вывод *DMGO* этой микросхемы. При условии, что данный не главный МП, выставивший запрос *DMR*, получил сигнал на вывод *DMGI*, он блокирует дальнейшее прохождение этого сигнала и выставляет сигнал *SACK*. Только после этого МП, захвативший шину адрес-данные, выставляет адрес и сигнал *BSY*, снимая при этом сигнал *DMR*, сигнал *SACK* снимается вместе с сигналами *DIN* и *DOUT* в зависимости от вида обмена.

Блокирование прохождения сигнала разрешения захвата прекращается после снятия сигнала *RPLY* пассивным устройством. Выставив следующий запрос *DMR*, МП может выполнять еще один цикл обмена, не сбрасывая сигнал *SACK*. Если главному МП необходимо захватить системную шину, то он блокирует выдачу сигнала *DMGO* на не главные устройства до окончания собственного цикла обмена и МП может выйти на шину лишь в случае отсутствия запросов *DMR* и *SACK* от не главных устройств. Если эти сигналы выставляются в течение цикла захвата главным МП шины, то сигналы запросов приостановят новый цикл захвата. Шина предоставляется не главному МП, сгенерировавшему запрос. Таким образом, наивысшим приоритетом в многопроцессор-

ной системе будет обладать устройство, непосредственно соединенное с главным МП по цепи *DMGO — DMGI*.

Остановимся на не менее важном механизме функционирования МП — прерывании программы. Внешнее устройство прерывает выполнение программы, например главного МП, запросом *VIRQ* на соответствующий вход МП (рис. 4.6). Каждое устройство, способное выставить запрос на прерывание, должно иметь разряд разрешения прерывания в своем регистре состояния. При установке в «1» этого разряда прерывание разрешается и МП переходит в режим обработки прерываний. Для этого он консервирует в стеке содержимое своего счетчика команд (регистр *R7*). Вслед за этим МП формирует сигнал *DIN* и через 200 нс сигнал разрешения прерывания *IAKO*. Устройство, вызвавшее прерывание, принимает на свой вход сигнал разрешения и блокирует дальнейшее его распространение как и в случае захвата шины. Для обеспечения однозначной реакции на прерывание устройство, инициализирующее запрос, помещает на шину *AD*0 ... *AD*7 адрес вектора прерывания. Это устройство выставляет сигнал *RPLY* и снимает *VIRQ*. МП принимает адрес вектора прерывания, снимает сигнал *DIN* и через 200 нс сигнал *IAKO*.

Таблица 4.4. Адреса внешних регистров

Адрес МП	*SEL*1	*SEL*2
00	177716	177714
01	177736	177734
10	177756	177754
11	177776	177774

Пассивное устройство снимает вектор прерывания и сигнал *RPLY*, МП загружает новое содержимое счетчика команд и регистра состояния из двух последовательных ячеек. Первая определяется адресом вектора прерывания. МП переходит к выполнению программы обслуживания прерывания. После завершения программы обслуживания МП выполняет команду возврата из прерывания (RTI). По ней МП восстанавливает содержимое счетчика команд и регистра состояния и продолжает выполнение прерванной программы.

Как отмечалось, в МП имеются два сигнала для программного обращения к внешним регистрам *SEL1* и *SEL2*. Адреса этих регистров зависят от номера МП при организации многопроцессорного включения (табл. 4.4).

Тип обмена с внешним регистром можно определить по сигналам *DIN* и *DOUT*. Сигнала *RPLY* в этом случае не требуется.

Для инициирования работы МП необходимо выполнить три фазы включения. В первой фазе при подаче питающего напряжения +5 В происходит начальная установка управляющих элементов. Он выставляет низкий уровень сигнала *INIT*. Фаза длится не менее 4 мс. Вторая фаза начинается с подачи сигнала *DCLO* и перехода сигнала *INIT* в «1» и продолжается до момента снятия

сигнала *DCLO* (не менее 70 мс). В третьей фазе начинается собственно работа МП. Происходит считывание *SEL*1 по адресу 177716. В счетчик команд заносятся содержимое *SEL*1 в старший байт, и нули — в младший байт. РСП загружается константой 340. В процессе выполнения программ он хранит информацию о текущем состоянии МП. В частности, 7-й разряд = «1» указывает на возможность внешнего устройства вызывать прерывание программы. Для удовлетворения требования прерывания 7-й разряд должен быть сброшен в «0». Четвертый разряд РСП называют

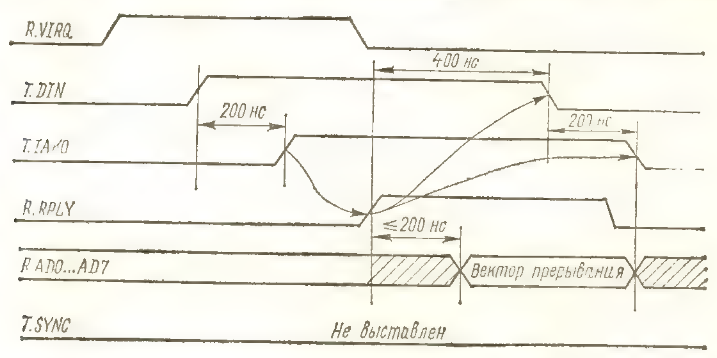


Рис. 4.6. Выполнение процедуры прерывания программы

Т-разрядом. Если он устанавливается в «1», то по завершении текущей команды будет вызвано прерывание с вектором 14. Использование этого разряда эффективно в отладочных режимах. Третий разряд РСП устанавливается в «1» при отрицательном результате, второй — при нулевом, первый разряд указывает на арифметическое переполнение, нулевой — на наличие переноса. Активным уровнем для всех признаков является «1».

4.2.2. Система команд МП

В МП используются безадресные, одноадресные и двухадресные команды. В одноадресных командах с 15-го разряда по 6-й включительно занимает код операции (КОП), с 5-го по 3-й — метод адресации, и со 2-го по 0-й — номер РОНа. Третий разряд определяет прямую или косвенную адресацию.

В двухадресных командах с 15-го по 12-й разряд устанавливается КОП, с 11-го по 9-й — метод адресации источника, с 8-го по 6-й — номер РОНа источника, с 5-го по 3-й — метод адресации приемника, со 2-го по 0-й — номер РОНа приемника. Для получения больших возможностей при написании программ пользователю предоставлена возможность описывать операнды несколькими методами адресации. Существуют четыре основных метода адресации, каждый из них в свою очередь может использо-

Таблица 4.5. Система команд микросхемы К1801ВМ1

Команда	Обозначение; код	Выработка признаков	Описание команды
Очистка, слово, байт	*CLR*; 0050*dd*	N, V, C — очищаются Z — устанавливается	$(dst) \leftarrow 000000$
Инвертирование, слово, байт	*COM*; 0051*dd*	V — очищается N, C, Z — устанавливаются	$(dst) \leftarrow (\overline{dst})$
Инкремент, слово, байт	*INC*; 0052*dd*	N, C, V — устанавливаются Z — не изменяется	$(dst) \leftarrow (dst) + 1$
Декремент, слово, байт	*DEC*; 0053*dd*	То же	$(dst) \leftarrow (dst) - 1$
Изменение знака, слово, байт	*NEG*; 0054*dd*	N, Z, V — устанавливаются C — очищается, если результат$=0$	$(dst) \leftarrow -(dst)$
Проверка, слово, байт	*TST*; 0057*dd*	N, Z — устанавливаются V, C — очищаются	$(dst) \leftarrow (dst)$
Арифметический сдвиг вправо, слово, байт	*ASR*; 0062*dd*	N, Z — загружаются младшим разрядом V, C — устанавливаются	$(dst) \leftarrow$ сдвинутое (dst) вправо арифметически на 1 разряд
Арифметический сдвиг влево, слово, байт	*ASL*; 0063*dd*	N, Z — устанавливаются $V = N \oplus Z$ C — старший разряд	$(dst) \leftarrow$ сдвинутое (dst) влево арифметически на 1 разряд
Циклический сдвиг вправо, слово, байт	*ROR*; 0060*dd*	N, Z — устанавливаются $V = N \oplus Z$ C — младший разряд	$(dst) \leftarrow$ сдвинутое (dst) вправо циклически на 1 разряд

Продолжение табл. 4.5

Команда	Обозначение; код	Выработка признаков	Описание команды
Циклический сдвиг влево, слово, байт	*ROL*; 0061*dd*	*N*, *Z* — устанавливаются $V = N \oplus Z$ *C* — старший разряд	(*dst*) ← сдвинутое (*dst*) влево циклически на 1 разряд (*C*) → младший разряд
Прибавление переноса, слово, байт	*ADC*; 0055*dd*	*N*, *V*, *C*, *Z* — устанавливаются	(*dst*) ← (*dst*) + (*C*)
Вычитание переноса, слово, байт	*SBC*; 0056*dd*	То же	(*dst*) ← (*dst*) — (*C*)
Расширение знака, слово	*SXT*; 0067*dd*	*N*, *C* — не изменяются *Z* — устанавливается *V* — очищается	Если $N = 1$, то 177777 → (*dst*) Если $N = 0$, то 000000 → (*dst*)
Перестановка байтов, слово	*SWAB*; 0003*dd*	*N*, *Z* — устанавливаются *V*, *C* — очищаются	Младший и старший байты меняются местами
Чтение слова-состояния, байт	*MFPS*; 1067*dd*	*N*, *Z* — устанавливаются *V* — очищается *C* — не изменяется	Восемь разрядов ССП пересылаются в указанную ячейку; (*dst*) ← ССП
Запись слова-состояния, байт	*MTRS*; 1064*dd*	Признаки не изменяются	Восемь разрядов указанного операнда замещают содержимое ССП
Пересылка, слово, байт	*MOV*; 1 *SS dd*	*N*, *Z* — устанавливаются *V* — очищается *C* — не изменяется	(*dst*) ← (*src*)

Продолжение табл. 4.5

Команда	Обозначение; код	Выработка признаков	Описание команды
Сравнение, слово, байт	*CMP*; 02 *SS dd*	N, Z, V — устанавливаются C — очищается, если был перенос	$(src) - (dst)$
Сложение, слово	*ADD*; 06 *SS dd*	Признаки устанавливаются	$(dst) \leftarrow (dst) + (src)$
Вычитание, слово	*SUB*; 16 *SS dd*	N, Z, V — устанавливаются C — очищается, если был перенос	$(dst) \leftarrow (dst) - (src)$
Исключающее ИЛИ, слово	*XOR*; 074 *R dd*	N, Z — устанавливаются V — очищается C — не изменяется	$(dst) \leftarrow (R) \oplus (dst)$
Ветвление безусловное	*BR*; 000400 *dd*	Признаки не изменяются	(СК) ← (СК) + 2xxx
Ветвление при условиях:			(СК) ← (СК) + 2xxx, если:
не нуль	*BNE*; 001000+xxx	Признаки не изменяются	$Z \neq 1$
нуль	*BEQ*; 001400+xxx		$Z = 1$
плюс	*BPL*; 100000+xxx		$N = 0$
минус	*BMI*; 100400+xxx		$N = 1$
нет переполнения	*BVC*; 102000+xxx		$V = 0$
есть переполнение	*BVS*; 102400+xxx		$V = 1$
нет переноса	*BCC*; 103000+xxx		$C = 0$
есть перенос	*BCS*; 103400+xxx		$C = 1$
$\geqslant 0$	*BGE*; 002000+xxx		$N \oplus V = 0$
< 0	*BLT*; 002400+xxx		$N \oplus V = 1$
> 0	*BGT*; 003000+xxx		$ZV (N \oplus V) = 0$
$\leqslant 0$	*BLE*; 003400+xxx		$ZV (N \oplus V) = 1$
больше	*BHI*; 101000+xxx		$CVZ = 0$
меньше или равно	*BLOS*; 101400+xxx		$CVZ = 1$
больше или равно	*BHIS*; 103000+xxx		$C = 0$
меньше	*BLO*; 103400+xxx		$C = 1$

Продолжение табл. 4.5

Команда	Обозначение; код	Выработка признаков	Описание команды
Безусловный переход Обращение к подпрограмме	*JMP*; 0001 *dd* *JSR*; 0004 *dd*	Признаки не изменяются	(СК) ← адрес *dst* (УС) ← (*R*) (*R*) ← (СК) (СК) ← адрес *dst*
Проверка разрядов, слово, байт	*BIT*; 03 *SS dd*	*N*, *Z* — устанавливаются *V* — очищается *C* — не изменяется	(*src*) $\wedge$ (*dst*)
Очистка разрядов, слово, байт	*BIC*; 04 *SS dd*	*N*, *Z* — устанавливаются, если старший разряд результата равен единице, в противном случае очищаются *V* — очищается *C* — не изменяется	(*dst*) ← $(\overline{src})$ $\wedge$ (*dst*)
Логическое сложение, слово, байт	*BIS*; 05 *SS dd*	*N* — устанавливается, если старший разряд результата установлен, иначе очищается *Z* — устанавливается, если все разряды результата очищены, в противном случае очищается *C* — не изменяется *V* — очищается	(*dst*) ← (*src*) $\wedge$ (*dst*)
Возврат из подпрограммы	*RTS*; 00020 *R*	Признаки не изменяются	(СК) ← (*R*) (*R*) ← (УС)

Продолжение табл. 4.5

Команда	Обозначение; код	Выработка признаков	Описание команды
Командное прерывание для системных программ	*EMT*; 104000—104377	Признаки загружаются из вектора прерывания	(УС) ← (РСП) (УС) ← (СК) (СК) ← (30) (РСП) ← (32)
Командное прерывание	*TRAP*; 104400—104777	То же	(УС) ← (РСП) (УС) ← (СК) (СК) ← (34) (РСП) ← (36)
Командное прерывание для ввода-вывода	*IOT*; 000004	То же	(УС) ← (РСП) (УС) ← (СК) (СК) ← (20) (РСП) ← (22)
Командное прерывание для отладки	*BPT*; 000003	То же	(УС) ← (РСП) (УС) ← (СК) (РСП) ← (16)
Возврат из прерывания 1	*RTI*; 000002	Признаки загружаются из стека	(СК) ← (УС) (РСП) ← (УС)
Возврат из прерывания 2	*RTT*; 000006	То же	(СК) ← (УС) (РСП) ← (УС)
Останов	*HLT*; 000000	Признаки загружаются из вектора прерывания	(177716) ← 000010 ∨ ∨ (177716) (177676) ← (ССП) (СК) ← (16002) (ССП) ← (160004)

Продолжение табл. 4.5

Команда	Обозначение; код	Выработка признаков	Описание команды
Ожидание	*WAIT*; 000001	Признаки не изменяются	Ожидание незамаскированного запроса
Сброс внешних устройств	*RESET*; 000005	То же	На выводе *INIT* формируется импульс длительностью 1120 периодов тактовой частоты
Пуск	*START*; 000012	Признаки загружаются из ячейки 176676	(177716) ← [177767 ∧ ∧ (177716)] (СК) ← (177674) (ССП) ← (177676)
Шаг	*S*; 000016		То же
Очистка N Z V C всех разрядов	 *CLN*; 000250 *CLZ*; 000244 *CLV*; 000242 *CLC*; 000241 *CCC*; 000257		Очистка: *N* *Z* *V* *C* *N*, *Z*, *V*, *C*
Установка: N Z V C всех разрядов	 *SEN*; 000270 *SFZ*; 000264 *SEV*; 000262 *SEC*; 000261 *SCC*; 000277		Установка: *N* *Z* *V* *C* *N*, *Z*, *V*, *C*
Пустая операция	*NOP*; 000240	Признаки не изменяются	Пустая операция
Умножение	*MUL*; 070 *R SS*	*N*, *Z*, *V* — устанавливаются *C* — очищается	$R \leftarrow (R) \times (src)$

ваться либо с прямой, либо с косвенной адресацией. Этими методами являются:

регистровый; при прямой адресации операнд указан в команде в виде номера (номеров) регистра;

автоинкрементный; при прямой адресации сначала выполняется действие, предусмотренное кодом операции, затем операнд увеличивается: на 1 — в случае байтовых операций, на 2 — в случае команд со словами;

автодекрементный; при прямой адресации вначале производят автоматическое уменьшение операнда, затем выполнение с ним действий, предусмотренных командой. Автоинкрементный и автодекрементный методы особенно эффективны при работе с массивами последовательно расположенных слов (байтов);

индексный; при прямой адресации исполнительный адрес вычисляется как сумма слова, содержащегося в следующей за командой ячейке, и содержимого выбранного РОНа.

В табл. 4.5 приведены команды МП. При описании команд используют следующие обозначения: АК — аккумулятор; *R* — регистр общего назначения; СК — счетчик команд; УС — указатель стека; *SS* — поле адресации источника; *dd* — поле адресации приемника; *src* — источник; *dst* — приемник; (*R*) — содержимое ячейки (регистра); *W* — знаковый разряд; *Z* — нулевой результат; *V* — переполнение; *C* — перенос; *NN* — смещение; →, (←) — пересылка.

4.2.3. Применение микросхем К1801ВМ1

На рис. 4.7 представлен пример построения системы с использованием *МП* К1801ВМ1. В систему входят следующие устройства: *системное ПЗУ*; *системное ОЗУ*; устройство управления *ОЗУ—УОЗУ*; регистр ввода-вывода *РВВ SEL1*.

Системное ПЗУ хранит программы по адресам: 160000—163777 — начальный пуск и отладочные режимы МП; 164000—172777 — резервная область; 173000—173777 — программы начальной загрузки с накопителя на гибких магнитных дисках (например, типа «Электроника ГМД-70») и программы тестирования.

Системное ОЗУ служит для временного хранения операндов и результатов операций. *РВВ SEL1* используют для задания режимов начального пуска, хранения флажков режимов.

Разряды 0, 1, 8, 9 ... 15 доступны только по чтению. Разряды 2 и 3 доступны по чтению и записи. Коды разрядов 0 и 1 определяют режимы начального пуска *МП*, а именно: 00 — через ячейки 24 и 26; 01 — пультовой режим работы *МП*; 10 — выход на программу начальной загрузки с диска; 11 — выход на программу тестирования.

Определение режима осуществляется в *системном ПЗУ*. Разряды 8 ... 15 определяют старший байт адреса ячейки, к кото-

рой осуществляется передача управления после включения питания. Младший байт адреса — нулевой.

Адрес начального пуска для системы полной конфигурации должен быть равен 160000. Разряды 2 и 3 являются флажками управления и признаками работы *МП*. Так, например, если 2-й разряд равен «1», *МП* работает в пультовом режиме, иначе — в режиме основной программы. Адрес регистра *РВВ SEL1* равен 177716, что позволяет радиально адресовать этот регистр сигналом *SEL1*.

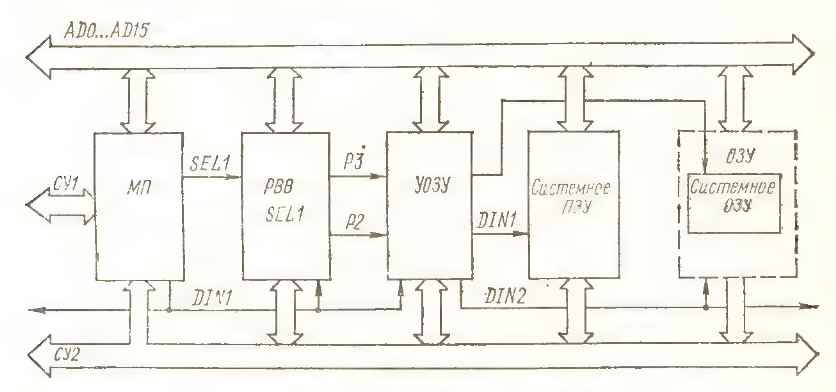


Рис. 4.7. Пример использования микросхемы К1801ВМ1

УОЗУ предназначено для преобразования адресов и регенерации *ОЗУ*. *УОЗУ* производит выборку *системного ОЗУ* из общего объема памяти. В пультовом режиме работы *УОЗУ* разрешает прохождение сигнала *DIN1* на *системное ПЗУ* и блокирует прохождение этого сигнала на другие устройства. В программном режиме работы *УОЗУ* блокирует прохождение сигнала *DIN1* на системное ПЗУ.

УОЗУ представляет возможность адресации внешних устройств в общем поле памяти. Для этого области памяти с 173000 до 173777, а также с 177600 до 177677 являются скрытыми (запрещенными) для основной программы.

Разряд *Р3* регистра *РВВ SEL1* (см. рис. 4.7) служит флажком сбойной ситуации типа «двойная ошибка» или «запрещенная команда». Единица в разряде *Р3* может быть использована для выработки сигнала *DCLO* для осуществления останова и перезапуска *МП* на программу обработки фатальных ситуаций.

Выход на программу обработки фатальных ситуаций при перезапуске *МП* осуществляется посредством анализа разряда *Р3*. На рис. 4.7 *СУ1*, *СУ2* — сигналы управления; *Р2*, *Р3* — разряды регистра *РВВ SEL1*.

Все времена даны для тактовой частоты f_{CLC} = 5 МГц.

В табл. 4.6 приведены статические параметры микросхемы К1801ВМ1.

Таблица 4.6. **Статические параметры микросхемы К1801ВМ1**

Параметр, единица измерения	Обозначение	Значение	Режим измерения
Выходное напряжение «0», В	U_{OL}	≤0,45	I_{OL} = 3,2 мА
Выходное напряжение «1», В	U_{OH}	≥2,45	I_{OH} = 0,2 мА
Входное напряжение «0», В	U_{IL}	≤0,5	I_{OL} = 3,2 мА
Входное напряжение «1», В	U_{IH}	≥2,4	I_{OH} = 0,2 мА
Ток потребления, мА	I_{CC}	≤220	—
Выходной ток утечки, мкА	I_{OZ}	≤50	—

4.3. МИКРОСХЕМА К1801РЕ1-000

Представляет постоянное запоминающее устройство (БИС ПЗУ) емкостью 65536 бит с организацией 4К × 16 разрядов. Предназначена для построения блоков системного ПЗУ, при этом возможно каскадное соединение с аналогичными БИС.

Условное графическое обозначение и структурная схема БИС ПЗУ даны соответственно на рис. 4.8 и 4.9. В табл. 4.7

Таблица 4.7. **Назначение выводов микросхемы К1801РЕ1-000**

Номер вывода	Назначение	Разряд	Обозначение	Тип
1	Чтение	—	*RE*	Вход
2	Ответ	—	*RPLY*	Выход
3	Синхронизация	—	*CE*	Вход
4	Шина адрес-данные	4	*AD4*	Вход-выход
5		5	*AD5*	
6		6	*AD6*	
7		7	*AD7*	
8		8	*AD8*	
9		9	*AD9*	
10		10	*AD10*	
11		11	*AD11*	
12	Общий	—	*GND*	—
13	Шина адрес-данные	3	*AD3*	Вход-выход
14		2	*AD2*	
15		1	*AD1*	
16		0	*AD0*	
17		12	*AD12*	
18		13	*AD13*	
19		14	*AD14*	
20		15	*AD15*	
21	Адрес	16	*A16*	Вход
22		17	*A17*	
23	Выбор микросхемы	—	*CS*	
24	Напряжение питания	—	U_{CC}	—

Примечание. Шина адрес-данные трехстабильная.

приведено назначение выводов микросхемы. В состав структурной схемы БИС ПЗУ входят: блок входных формирователей *БВхФ*, адресные регистры *РА1* и *РА2*, блок формирования сигнала выборки микросхемы *БФВМ*, дешифратор *ДШ*, матрица постоянного запоминающего устройства *ПЗУ*, блок формирователей *Ф*, выходной регистр *Вых Рг*, выходные усилители *У*, блок формирования сигнала синхронизации *БФСС*, блок управления выходными сигналами *БУ*.

БВхФ предназначен для приема внешних сигналов *CE*, *CS*, *A16*, *A17*, а также для формирования сигнала *RPLY* и внутри-

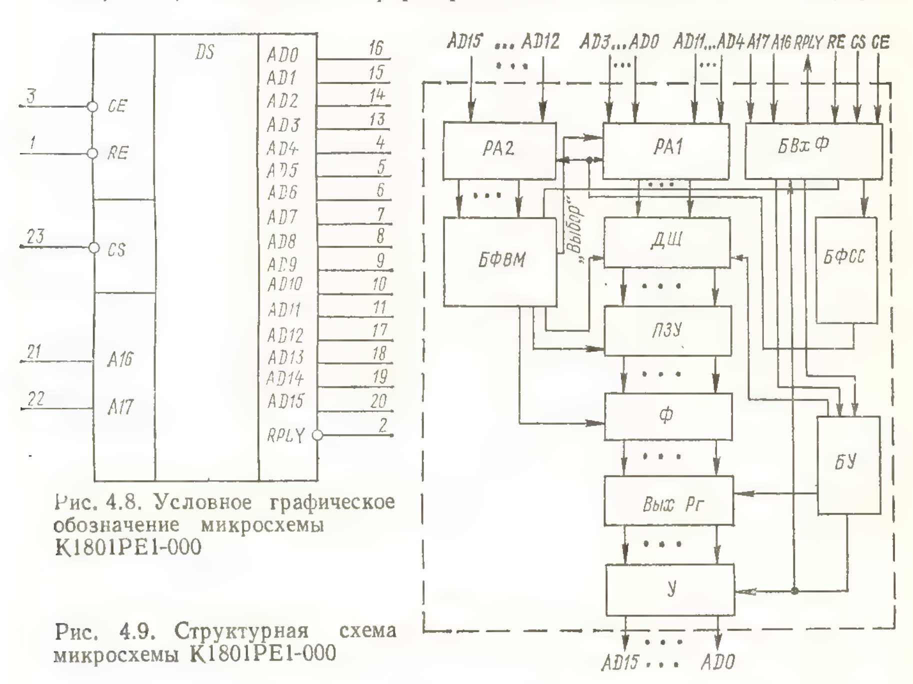


Рис. 4.8. Условное графическое обозначение микросхемы К1801РЕ1-000

Рис. 4.9. Структурная схема микросхемы К1801РЕ1-000

схемных управляющих сигналов. Один из внутренних сигналов запускает *БФСС* и инициализирует выработку сигнала синхронизации на ДШ.

Адрес считываемой ячейки поступает по шине *AD* на адресные регистры *РА1* и *РА2*.

На регистре *РА1* адрес запоминается и выдается на *ДШ* при наличии внутреннего сигнала «Выбор» из блока *БФВМ*. Последний появляется при активизации данной БИС в каскадном соединении нескольких аналогичных схем. Это определяется старшими *AD12 ... AD15* разрядами адреса. *БФВМ* формирует сигналы управления для *ДШ*, *ПЗУ* и *Ф*.

Блок формирователей управляет выдачей данных из *Вых Рг* через *У*, а также влияет на сигнал *RPLY*.

В выбранной микросхеме адрес *AD*0 ... *AD*11 дешифрируется на *ДШ* и поступает на *ПЗУ*.

Информация поступает через *Ф*, *Вых Рг* на *У* и выдается на выводы *AD0* ... *AD15*.

При построении БИС ПЗУ был использован принцип временного мультиплексирования, что обусловлено, в первую очередь, особенностями микроЭВМ «Электроника-60».

БИС ПЗУ управляют четыре сигнала: *CE* — обеспечивает запись адреса во входные адресные регистры микросхемы *RE* — выдачу данных на магистраль *AD* при наличии *CE* = «0». Выданная информация сопровождается сигналом *RPLY* = «0». Сигнал *CS* осуществляет выборку микросхемы.

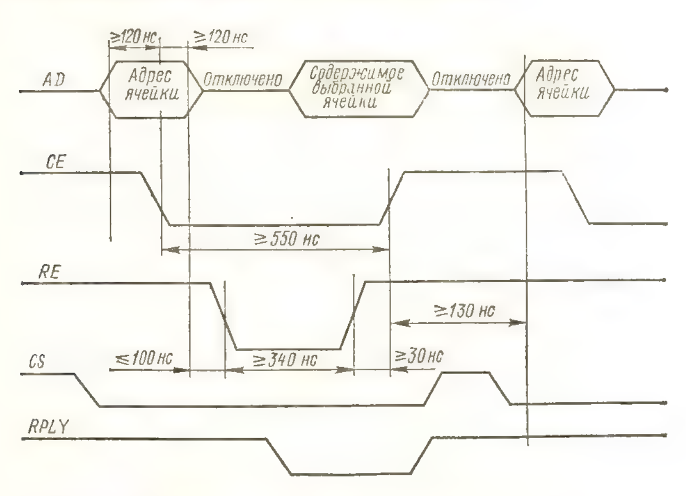


Рис. 4.10. Временна́я диаграмма работы микросхемы К1801РЕ1-000

Разряды *AD12* ... *AD15* позволяют каскадно подключать до 16 микросхем *ПЗУ*. Возможна байтовая выборка. В этом случае код адреса подается на входы *AD1* ... *AD12* (*AD0* не участвует в выборке адреса). Код микросхемы поступает на входы *AD13* ...

Таблица 4.8. Статические параметры микросхемы К1801РЕ1

Параметр, единица измерения	Обозначение	Значение	Режим измерения, мА
Выходное напряжение «0», В	U_{OL}	≤0,5	$I_{OL} = 3,2$
Выходное напряжение «1», В	U_{OH}	≥2,4	$I_{OH} = 0,4$
Мощность потребления в режиме хранения, мВт	P_{CCS}	≤150	—
Мощность потребления в режиме обмена, мВт	P_{CC}	≤250	—
Входной ток «1», мкА	I_{IH}	≤1,0	—

AD15. Информация считывается с линий *AD0* ... *AD15*. В этом случае можно адресоваться к восьми микросхемам.

Разряды *A16* и *A17* обеспечивают дополнительные возможности по выборке БИС и считыванию полноразрядных слов. Для этого необходимо на вход *A16* подать «0», а на вход *A17* — «1», Временная диаграмма работы БИС ПЗУ изображена на рис. 4.10. Основные электрические параметры при напряжении питания U_{CC} = = +5 В (±5 %) приведены в табл. 4.8.

4.4. МИКРОСХЕМА К1801ВП1-030

Предназначена для использования в блоках ОЗУ, выполненных на микросхемах К565РУ3 с динамическим хранением информации. Применяется в качестве устройства управления обменом информации. Микросхема осуществляет:

прием, хранение и преобразование адреса ОЗУ;

отработку интерфейсных сигналов системной магистрали ЭВМ типа «Электроника-60»;

отработку сигналов синхронизации ОЗУ и буферного регистра данных;

регенерацию памяти;

выработку вспомогательного сигнала блокирования *LOCK* при обращении в область внешних устройств (старшие 4К слов). *LOCK* предназначен для использования данной БИС совместно с процессором на базе микросхемы К1801ВМ1.

Условное графическое обозначение микросхемы приведено на рис. 4.11, структурная схема дана на рис. 4.12. В табл. 4.9 раскрыто назначение выводов.

Рис. 4.11. Условное графическое обозначение К1801ВП1-030

В структурную схему входят следующие устройства: усилители мощности *УМ*, служат для развязки и увеличения мощности сигналов; счетчик адресов регенерации *СЧАР*, состоит из 7-разрядного делителя тактовой частоты (разряды 0 ... 6) и собственно 7-разрядного счетчика адресов регенерации (разряды 7 ... 13), которые выдаются непосредственно через мультиплексор адресов *МПА* на выводы *A0* ... *A6*; буферный регистр адреса *БРА* (разряды 0 ... 15), служит для хранения адреса, подаваемого по системной магистрали.

Разряды адреса (1 ... 7) выдаются в фазе выдачи адреса строки через *МПА* на выводы *A0* ... *A6* в режиме обмена. Разряды

AD8 ... AD14 выдаются в фазе выдачи адреса колонки и используются компаратором адресов *КМПА* для выработки сигнала *LOCK*. Разряд 0 используется блоком синхронизации *БСЗ* для определения номера байта при выполнении процедуры записи байта.

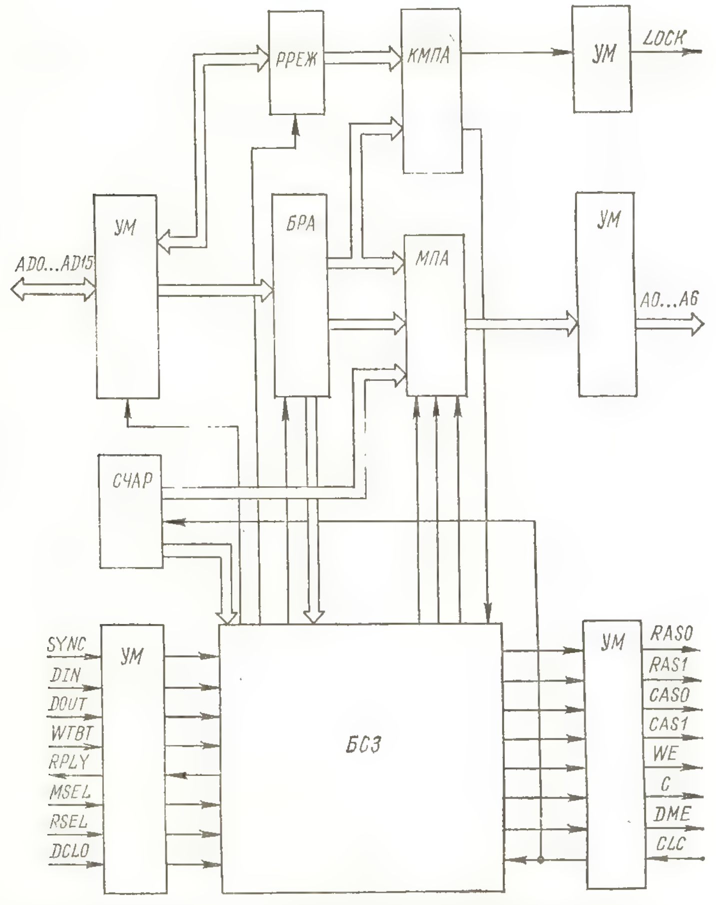


Рис. 4.12. Структурная схема микросхемы К1801ВП1-030

Запись информации в *БРА* происходит сигналом «1», защелкивание информации — сигналом «0».

МПА служит для раздельной во времени выдачи адреса ОЗУ в виде 7-разрядного адреса строки и 7-разрядного адреса колонки в циклах обмена, а также 7-разрядного адреса регенерации.

Вспомогательный регистр режима *РРЕЖ* служит для дублированного хранения признаков режимов работы процессора К1801ВМ1 «Останов» и «Расширенная арифметика». Разрядность

Таблица 4.9. **Назначение выводов микросхемы К1801ВП1-030**

Номер вывода	Назначение	Разряд	Обозначение	Тип
1	Шина адрес-данные	5	*AD*5	Вход-выход
2		4	*AD*4	
3		3	*AD*3	
4		2	*AD*2	
5		1	*AD*1	
6		14	*AD*14	
7		0	*AD*0	
8	Чтение данных	—	*DIN*	Вход
9	Запись данных	—	*DOUT*	
10	Синхронизация	—	*CLC*	
11	Сопровождение адреса строки	0	*RAS*0	Выход
12	Блокировка	—	*LOCK*	
13	Ответ	—	*RPLY*	
14	Стробирование записи в буферный регистр	—	*C*	
15	Выборка данных	—	*DME*	
16	Сопровождение адреса строки	1	*RAS*1	
17	Выборка регистра режима	—	*RSEL*	
18	Сопровождение записи	—	*WE*	
19	Сопровождение адреса колонки	0	*CAS*0	
20		1	*CAS*1	
21	Общий	—	*GND*	—
22	Шина адреса	0	*A*0	Выход
23		1	*A*1	
24		2	*A*2	
25		3	*A*3	
26		4	*A*4	
27		5	*A*5	
28		6	*A*6	
29	Запись-байт	—	*WTBT*	Вход
30	Авария источника питания	—	*DCL*0	
31	Шина адрес-данные	15	*AD*15	Вход-выход
32	Обмен	—	*SYNC*	Вход
33	Выбор памяти	—	*MSEL*	
34	Шина адрес-данные	13	*AD*13	Вход-выход
35		12	*AD*12	
36		11	*AD*11	
37		10	*AD*10	
38		9	*AD*9	
39		8	*AD*8	
40		7	*AD*7	
41		6	*AD*6	
42	Напряжение питания	—	U_{cc}	—

РРЕЖ — 2 бита. По записи и чтению доступен разрядам магистрали соответственно *AD2* и *AD3*. Режиму «Останов» соответствует значение «1» в разряде 2 *РРЕЖ*, режиму «Расширенная арифметика» — значение «1» в разряде 1 *РРЕЖ*.

КМПА вырабатывает сигнал блокировки *LOCK*, который служит для выборки области адресов из системного ПЗУ процессора и блокировки этих же областей в адресном пространстве внешних устройств. Независимо от состояния разрядов *РРЕЖ* (2—3) сигнал *LOCK* вырабатывается, если текущий адрес находится в диапазоне адресов от $173000_{(8)}$ до $173777_{(8)}$. В режиме «Останов» *LOCK* вырабатывается при нахождении адреса обращения в диапазоне от $160000_{(8)}$ до $163777_{(8)}$, в режиме «Расширенная арифметика» — в диапазоне от $160000_{(8)}$ до $173777_{(8)}$.

КМПА вырабатывает признак «Свой» для блока синхронизации *БСЗ* при обращении по адресу в диапазоне от $177000_{(8)}$ до $177677_{(8)}$, который позволяет осуществить запуск *БСЗ* при отсутствии сигнала *MSEL*. *БСЗ* вырабатывает сигналы управления внутренними узлами микросхемы, внешним ОЗУ, внешним буферным регистром данных.

При работе микросхемы может выполняться один из пяти циклов функционирования: «Регенерация», «Чтение», «Запись», «Блокировка», «Установка».

«Регенерация» (восстановление) информации в памяти динамического типа производится по принципу: один цикл регенерации по одному адресу строки в течение 15,6 мкс. Таким образом, за 2 мс происходит полная регенерация информации в памяти по всем адресам. В *БСЗ* введена схема подсинхронизации запуска текущего цикла регенерации к окончанию цикла обмена информацией, что делает память динамического типа максимально «прозрачной» для процессора.

В случае отсутствия циклов обмена с памятью в течение 8 мкс после получения *БСЗ* запроса на регенерацию очередной цикл регенерации производится принудительно без синхронизации.

Сигналы *RAS0* и *RAS1* вырабатываются одновременно.

На рис. 4.13 приведена временна́я диаграмма работы БИС УОЗУ.

Цикл «Чтение» вызывает сигнал *SYNC*, по которому фиксируются адрес обращения (*AD0* ... *AD15*) в БРА, сигналы *MSEL* и *WTBT*, а также отсутствие очередного цикла «Регенерация». Сигналы *DME* и *RPLY* вырабатываются только при наличии сигнала *DIN*. Сигнал *RAS0* вырабатывается, если *AD15* равен «1» в фазе выдачи адреса, а *RAS1* — «0». Сигналы *CAS0* и *CAS1* вырабатываются одновременно.

Цикл «Запись» вызывает сигнал *SYNC*, по которому фиксируются адрес обращения *AD0* ... *AD15*, сигналы *MSEL*, *WTBT* в элементах хранения *БСЗ*, сигнал *DOUT*, а также отсутствие очередного цикла «Регенерация».

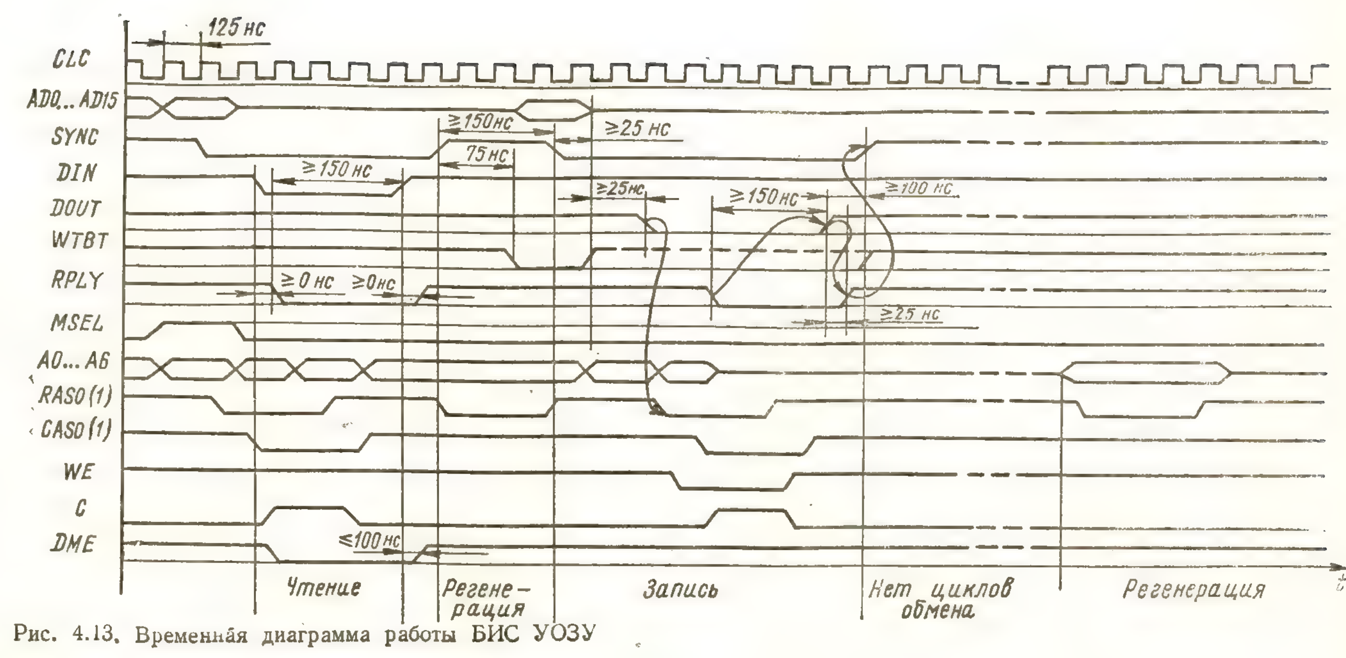


Рис. 4.13. Временная диаграмма работы БИС УОЗУ

Таблица 4.10. Статические параметры микросхемы К1801ВП1-030

Параметр, единица измерения	Обозначение	Значение	Режим измерения
Выходное напряжение «0», В	U_{OL}	$\leq 0,5$	$U_{CC} = 4,75$ В $I_{OL} = 3,2$ мА
Выходное напряжение «1», В	U_{OH}	$\geq 2,4$	$U_{CC} = 4,75$ В $I_{OH} = 1,0$ мА
Ток утечки на входе, мкА	I_{IZ}	≤ 10	$U_{CC} = 5,25$ В $U_{IL} = 0$ В $U_{IH} = U_{CC}$
Ток потребления	I_{CC}	≤ 200	$U_{CC} = 5,25$ В $U_{LL} = 0$ В $U_{IH} = U_{CC}$

Наличие сигнала *WTBT* в фазе выдачи адреса является признаком записи адреса, а в фазе выдачи данных — признаком записи байта. При записи байта *БСЗ* вырабатывает сигнал *CAS0*, если *AD0* равен «1» в фазе адреса, в противном случае вырабатывается сигнал *CAS1*.

При записи слова вырабатываются сигналы *CAS0* и *CAS1* одновременно.

Сигнал блокирования *LOCK* вырабатывается в фазе выдачи адреса в цикле «Блокировка».

Задержка появления сигнала *LOCK* относительно момента выдачи адреса на системную магистраль не более 100 нс.

Микросхема переходит в состояние готовности за время, соответствующее длительности прохождения семи синхронизирующих импульсов *CLC*.

Сигналом *DCLO* производится установка СЧАР в нулевое состояние, РРЕЖ — в режим «Останов».

Динамические параметры микросхемы приведены на временно́й диаграмме рис. 4.13, статические — в табл. 4.10.

4.5. МИКРОСХЕМА К1801ВП1-033

Микросхема К1801ВП1-033 — это многофункциональное устройство, которое может работать в режимах: интерфейса накопителя на гибких магнитных дисках, контроллера интерфейса параллельного ввода-вывода, контроллера байтового параллельного интерфейса ввода-вывода ИРПР.

Микросхема может использоваться совместно с микросхемой К1801ВП1-034 для организации интерфейсного устройства 16-разрядного программированного параллельного ввода-вывода и интерфейсного устройства байтового параллельного ввода-вывода,

Таблица 4.11. **Назначение выводов микросхемы К1801ВП1-033 в режиме интерфейса накопителя на гибких магнитных дисках**

Номер вывода	Назначение	Разряд	Обозначение	Тип
1	Выбор режима	3	*RC*3	Вход
2		0	*RC*0	
3		1	*RC*1	
4		2	*RC*2	
5		4	*RC*4	
6		5	*RC*5	
7	Сдвиг данных	—	*SHFT*	
8	Вывод данных	—	*OUT*	
9	Шина адрес-данные	0	*AD*0	Вход-выход
10		1	*AD*1	
11		2	*AD*2	
12		3	*AD*3	
13		4	*AD*4	
14		5	*AD*5	
15		6	*AD*6	
16		7	*AD*7	
17		8	*AD*8	
18		9	*AD*9	
19		10	*AD*10	
20		11	*AD*11	
21	Общий	—	*GND*	—
22	Шина адрес-данные	12	*AD*12	Вход-выход
23	Внешнее устройство	—	*BS*	Вход
24	Шина адрес-данные	14	*AD*14	Вход-выход
25		15	*AD*15	
26	Вход данных	—	*DI*	Вход
27	Выход данных	—	*D*0	Выход
28	Пуск	—	*RUN*	
29	Начальная установка	—	*SET*	
30	Ошибка	—	*ERR*	Вход
31	Завершено	—	*DONE*	
32	Требование передачи	—	*TR*	
33	Разрешение прерывания	—	*IAKI*	
34	Сброс	—	*INIT*	
35	Запрос прерывания	—	*VIRQ*	Выход
36	Разрешение прерывания	—	*IAKO*	
37	Вывод данных	—	*DOUT*	Вход
38	Ввод данных	—	*DIN*	
39	Ответ	—	*RPLY*	Выход
40	Запись-байт	—	*WTBT*	Вход
41	Обмен	—	*SYNC*	
42	Напряжение питания	—	U_{cc}	—

Таблица 4.12. **Назначение выводов микросхемы К1801ВП1-033 в режиме контроллера интерфейса параллельного ввода-вывода**

Номер вывода	Назначение	Разряд	Обозначение	Тип
1	Выбор режима	3	*RC*3	Вход
2		0	*RC*0	
3		1	*RC*1	
4		2	*RC*2	
5	Задержка ответа	—	*RD*0	Выход
6		—	*RD*1	Вход
7	Выход регистра состояния	0	*RC*0	Выход
8		1	*RC*1	
9	Шина адрес-данные	0	*AD*0	Вход-выход
10		1	*AD*1	
11		2	*AD*2	
12		3	*AD*3	
13		4	*AD*4	
14		5	*AD*5	
15		6	*AD*6	
16		7	*AD*7	
17		8	*AD*8	
18		9	*AD*9	
19		10	*AD*10	
20		11	*AD*11	
21	Общий	—	*GND*	—
22	Шина адрес-данные	12	*AD*12	Вход-выход
23	Внешнее устройство	—	*BS*	Вход
24	Шина адрес-данные	14	*AD*14	Вход-выход
25		15	*AD*15	
26	Вывод старшего байта	—	*BIR*	Выход
27	Вывод данных	—	*DIR*	
28		—	*NDR*	
29	Вывод младшего байта	—	*BOR*	
30	Требование *B*	—	*REQB*	Вход
31	Чтение выходного регистра	—	*ORR*	Выход
32	Требование *A*	—	*REQA*	Вход
33	Разрешение прерывания	—	*IAKI*	
34	Сброс	—	*INIT*	
35	Запрос прерывания	—	*VIRQ*	Выход
36	Разрешение прерывания	—	*IAKO*	
37	Вывод данных	—	*DOUT*	Вход
38	Ввод данных	—	*DIN*	
39	Ответ	—	*RPLY*	Выход
40	Запись-байт	—	*WTBT*	Вход
41	Обмен	—	*SYNC*	
42	Напряжение питания	—	U_{cc}	—

Таблица 4.13. **Назначение выводов микросхемы К1801ВП1-033 в режиме контроллера байтового параллельного интерфейса ввода-вывода ИРПР**

Номер вывода	Назначение	Разряд	Обозначение	Тип
1	Требование	—	*REQ*	Вход
2	Выбор режима	0	*RC*0	
3		1	*RC*1	
4		2	*RC*2	
5	Готовность приемника	—	*AO-A*	
6	Запрос приемника	—	*AC-A*	
7	Запрос источника	—	*AC-S*	Выход
8	Строб приемника	—	*SC-A*	
9	Шина адрес-данные	0	*AD*0	Вход-выход
10		1	*AD*1	
11		2	*AD*2	
12		3	*AD*3	
13		4	*AD*4	
14		5	*AD*5	
15		6	*AD*6	
16		7	*AD*7	
17		8	*AD*8	
18		9	*AD*9	
19		10	*AD*10	
20		11	*AD*11	
21	Общий	—	*GND*	—
22	Шина адрес-данные	12	*AD*12	Вход-выход
23	Внешнее устройство	—	*BS*	Вход
24	Шина адрес-данные	14	*AD*14	Вход-выход
25		15	*AD*15	
26	Строб источника	—	*SC-S*	Вход
27	Ввод-данных	—	*IN*	Выход
28	Вывод данных	—	*OUT*	
29	Начальная установка	—	*SET*	
30	Ошибка	—	*ERR*	Вход
31	Завершено	—	*DONE*	
32	Требование передачи	—	*TR*	
33	Разрешение прерывания	—	*IAKI*	
34	Сброс	—	*INIT*	
35	Запрос прерывания	—	*VIRQ*	
36	Разрешение прерывания	—	*IAKO*	Выход
37	Вывод данных	—	*DOUT*	Вход
38	Ввод данных	—	*DIN*	
39	Ответ	—	*RPLY*	Выход
40	Готовность источника	—	*SO-S*	Вход
41	Обмен	—	*SYNC*	
42	Напряжение питания	—	*Ucc*	—

а также как самостоятельное интерфейсное устройство накопителя на гибких магнитных дисках.

Обозначение и выводы для трех режимов работы приведены в табл. 4.11 ... 4.13. Условные обозначения для режимов даны на рис. 4.14 ... 4.16. При описании структурных схем были использованы следующие сокращения: БП — блок прерываний, БСЗ —

Рис. 4.14. Условное графическое обозначение микросхемы К1801ВП1-033 в режиме интерфейса НГМД

Рис. 4.15. Условное графическое обозначение микросхемы К1801ВП1-033 в режиме контроллера интерфейса параллельного ввода-вывода

Рис. 4.16. Условное графическое обозначение микросхемы К1801ВП1-033 в режиме контроллера байтового параллельного интерфейса ввода-вывода

блок синхронизации, БК — блок канала, БРР — блок режима работы, КМПА — компаратор адреса, РС — регистр состояния, БСЗ В/В — блок синхронизации ввода-вывода, РКС — регистр команд и состояний, СРД — сдвиговый регистр данных.

4.5.1. Режим накопителя на гибких магнитных дисках

Установка микросхемы в режим накопителя на гибких магнитных дисках (НГМД) производится подачей «1» на выводы *RC3 ... RC0*. Информация между микропроцессором и контроллером НГМД передается микросхемой посредством *РКС* и *СРД*. Эти регистры считываются и загружаются программно. Структура

микросхемы в этом режиме приведена на рис. 4.17. Ниже дана расшифровка разрядов регистра *РКС*:

G0 (разряд 0) — инициализация приема контроллером НГМД команды;

F3 ... F1 (разряды 3 ... 1) — команда интерфейса НГМД;

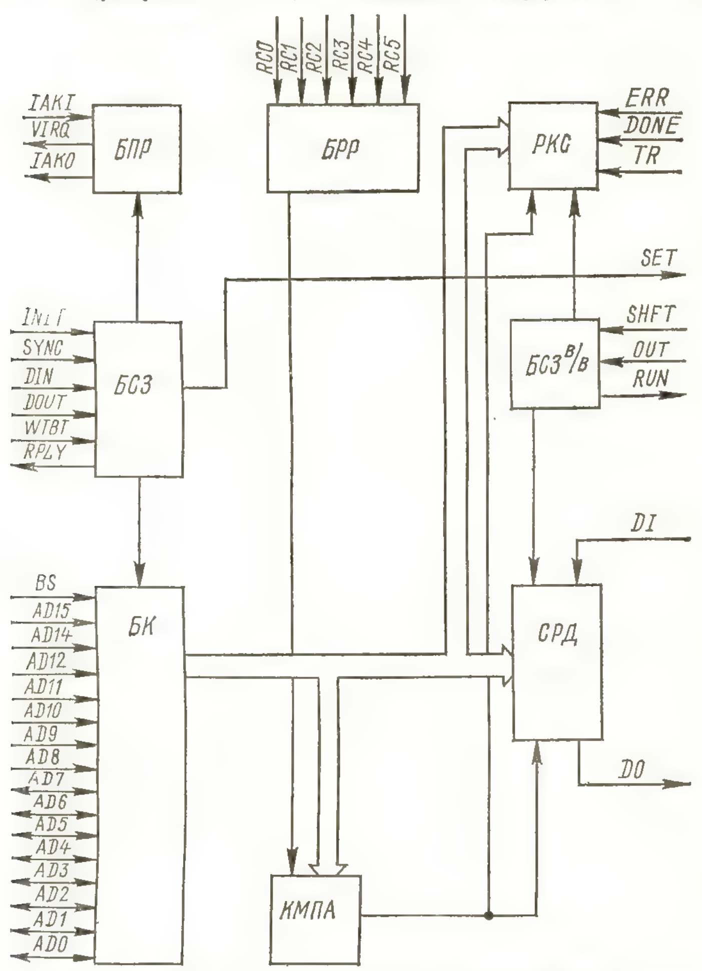


Рис. 4.17. Структурная схема микросхемы К1801ВП1-033 в режиме интерфейса НГМД

US (разряд 4) — выбор одного из двух дисководов для выполнения требуемой команды;

DONE (разряд 5) — завершение выполнения команды;

IEE (разряд 6) — разрешение прерывания (устанавливается программно для осуществления прерывания по завершении выполнения команды, сбрасывается сигналом *INIT*);

Таблица 4.14. **Команды интерфейса НГМД**

Код команды			Действие команды
F3	F2	F1	
0	0	0	Запись в буфер
0	0	1	Чтение буфера
0	1	0	Запись сектора
0	1	1	Чтение сектора
1	0	0	Не используется
1	0	1	Чтение регистра обмена и состояния
1	1	1	Запись сектора с меткой
1	1	1	Чтение регистра ошибки

TR (разряд 7) — требование записи-считывания данных через регистр *СРД*;

INIT (разряд 14) — начальная установка (устанавливается программно для приведения НГМД в исходное состояние);

ERR (разряд 15) — ошибка (устанавливается, если в процессе выполнения команды произошла ошибка).

Разряды 13 ... 8 не используются. Список команд представлен в табл. 4.14.

CPD загружается младшим байтом шины адрес-данные (системной магистрали).

Ниже приводятся интерфейсные сигналы контроллера НГМД:

SET — вырабатывается микросхемой для приведения механизма и электронной части НГМД в исходное состояние;

DONE — устанавливается НГМД для индикации того, что текущая команда выполнена;

RUN — устанавливается микросхемой для инициирования передачи команды или байта данных;

OUT — вырабатывается НГМД для указания направления передачи байта информации;

TR — устанавливается НГМД для индикации того, что он готов принять (передать) байт информации;

DI — предназначен для приема последовательной посылки информации;

DO — предназначен для передачи последовательной посылки информации;

SHET — предназначен для синхронизации приема (передачи) последовательной посылки информации (длительность отрицательного импульса не менее 200 нс, период — 1000 нс);

ERR — устанавливается НГМД при обнаружении ошибки, при этом прекращается выполнение текущей команды и устанавливается сигнал *DONE*.

Изменяя сигналы на выводах *RC4* и *RC5*, можно переадресовать регистры микросхем. Соответственно меняются адреса век-

Таблица 4.15. **Адресация регистров микросхемы К1801ВП1-033 сигналами** *RC*4 **и** *RC*5

Сигнал	Уровень сигнала	Адрес		
		РКС	СРД	Вектор прерывания
*RC*4 *RC*5	Низкий	177170	177172	264
*RC*4 *RC*5	Высокий Низкий	177174	177176	270
*RC*4 *RC*5	Низкий Высокий	177200	177202	274
*RC*4 *RC*5	Высокий	XXXXX0	XXXXX2	XXX

торов прерываний. Адреса регистров и векторов прерываний, соответствующие комбинациям уровней напряжений на выводах *RC4*, *RC5*, представлены в табл. 4.15.

При низком уровне сигнала *DONE* запись в *РКС* команды с «1» в нулевом разряде вызывает установку сигнала *RUN*, который инициирует НГМД на прием команды. НГМД снимает сигнал *DONE* и выдает на линию *SHFT* серию из восьми импульсов. После снятия сигнала *DONE* снимается сигнал *RUN*, а серия импульсов синхронизирует выдачу команды в последовательном коде на вывод *DO*. В зависимости от принятого кода команды НГМД устанавливает сигналы *OUT* и *TR*. При установленном сигнале *TR* в зависимости от состояния сигнала обращение к *СРД* вызывает установку сигнала *RUN* (который сбрасывается после снятия сигнала *TR*) и появление серии импульсов на выводе *SHFT* (восемь для синхронизации адреса сектора и дорожки, семь для синхронизации данных). По окончании выполнения команды устанавливается сигнал *DONE*, который используется для работы по прерыванию. Запрос прерывания *VIRQ* возникает при появлении сигнала *DONE* и при наличии в *РКС* разрешения прерывания *IE*.

4.5.2. Режим контроллера интерфейса параллельного ввода-вывода

Установка микросхемы в этот режим производится подачей определенных уровней напряжéния на выводы *RC*0 ... *RC*3. Помимо выбора режима работы микросхемы комбинациями сигналов осуществляют переадресацию регистров и векторов прерываний. Адреса регистров и векторов прерываний, соответствующие определенным комбинациям сигналов на выводах *RC*0 ... *RC*3, представлены в табл. 4.16.

Таблица 4.16. Адресация регистров микросхемы К1801ВП1-033 сигналами *RC*0 ... *RC*3

Сигнал	Уровень сигнала	РС	РИ	РП	Вектор прерывания	
					А	*В*
*RC*3 *RC*0 *RC*1 *RC*2	Низкий Высокий Низкий Высокий	167770	167772	167774	300	304
*RC*3 *RC*0 *RC*1 *RC*2	Высокий Высокий Низкий Высокий	167760	167762	167764	310	314
*RC*3 *RC*0 *RC*1 *RC*2	Низкий Низкий Высокий Высокий	167750	167752	167754	320	324
*RC*3 *RC*0 *RC*1 *RC*2	Высокий Низкий Высокий Высокий	167740	167442	167744	330	334
*RC*3 *RC*0 *RC*1 *RC*2	Низкий Высокий Высокий Высокий	XXXXX0	XXXXX2	XXXXX4	XX0	XX4

Микросхема осуществляет прием и передачу информации с помощью регистра состояния *РС*, регистра приемника РП и регистра источника РИ. Структурная схема микросхемы в этом режиме представлена на рис. 4.18. РП и РИ не входят в состав микросхемы и могут быть выполнены на двух микросхемах К1801ВП1-034. Микросхема К1801ВП1-033 в режиме контроллера интерфейса параллельного ввода-вывода содержит регистр состояния *РС*, компаратор адреса *КМПА*, блок прерываний *БПР*, блок синхронизации ввода-вывода *БСЗ В/В*, блок синхронизации *БСЗ*, блок канала *БК*. В режиме контроллера интерфейса параллельного ввода-вывода микросхема может обрабатывать запросы на прерывание *А* и *В*.

Расшифровка разрядов РС и сигналов управления РП и РИ:

CSRI, *CSRO* (разряды 0 и 1) — имитация прерывания в режиме автономной проверки;

REQA (разряд 7) — требование прерывания *A* (только для чтения);

REQB (разряд 15) — требование прерывания *B* (только для чтения);

IEA (разряд 6) — разрешение прерывания *A*, сбрасывается сигналом *INIT* (для чтения и записи);

IEB (разряд 5) — разрешение прерывания *B*, сбрасывается сигналом *INIT* (для чтения и записи).

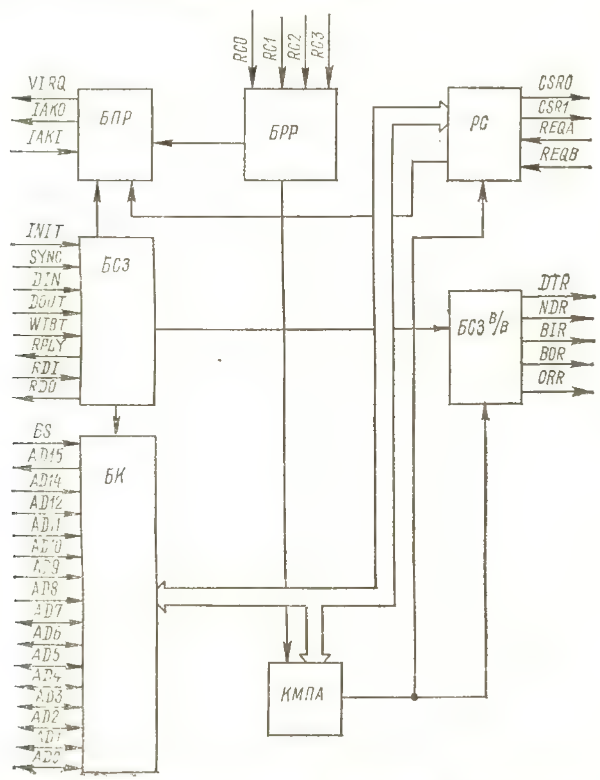


Рис. 4.18. Структурная схема микросхемы К1801ВП1-033 в режиме контроллера интерфейса параллельного ввода-вывода

Сигналы управления РП и РИ:

DTR — вырабатывается микросхемой при чтении РП;

NDR — вырабатывается микросхемой при записи в РИ;

BIR — вырабатывается микросхемой при записи старшего байта в РИ;

BOR — вырабатывается микросхемой при записи младшего байта в РИ;

ORR — вырабатывается микросхемой при чтении РИ;

RDO — используется для цепочки формирования сигнала *RPLY*;

RDI — используется для цепочки формирования сигнала *RPLY*.

С помощью RC-цепочки, включенной между выводами *RDI* и *RDO*, можно увеличивать длительность импульсов *DTR* и *NDR*.

При работе микросхемы по прерыванию запрос прерывания *VIRQ* возникает при появлении хотя бы одного из сигналов *REQA* или *REQB* при наличии в регистре состояния соответствующего разрешения прерывания *IEA*, *IEB*. При одновременном появлении сигналов *REQA* и *REQB* более высокий приоритет имеет требование *B*.

4.5.3. Режим контроллера байтового параллельного интерфейса ввода-вывода

Установка микросхемы в этот режим производится подачей определенных сигналов на входы *RC0 ... RC2*. Помимо выбора режима работы микросхемы комбинации сигналов на входах *RC0 ... RC2* осуществляют переадресацию регистров и векторов прерываний. Адреса регистров и векторов прерываний, соответствующие определенным комбинациям сигналов на входах

Таблица 4.17. Адресация регистров микросхемы К1801ВП1-033 сигналами *RC0 ... RC2*

Сигнал	Уровень сигнала	Адрес				Вектор прерывания
		РСИ	РП	РСП	РИ	
RC0 *RC1* *RC2*	Низкий Низкий Низкий	—	—	177514	177516	200
RC0 *RC1* *RC2*	Высокий Низкий Низкий	177560	177562	177564	177566	060
RC0 *RC1* *RC2*	Низкий Высокий Низкий	177550	177552	177554	177556	070
RC0 *RC1* *RC2*	Высокий Высокий Низкий	177270	177272	177274	177276	170
RC0 *RC1* *RC2*	Низкий Низкий Высокий	XXXXX0	XXXXX2	XXXXX4	XXXXX6	XX0

RC0 ... RC2, представлены в табл. 4.17. Структура микросхемы в этом режиме представлена на рис. 4.19.

Микросхема осуществляет прием и передачу информации с помощью регистров состояния источника РСИ и приемника РСП, регистров источника РИ и приемника РП. Регистры источника выполнены на одной микросхеме К1801ВП1-034.

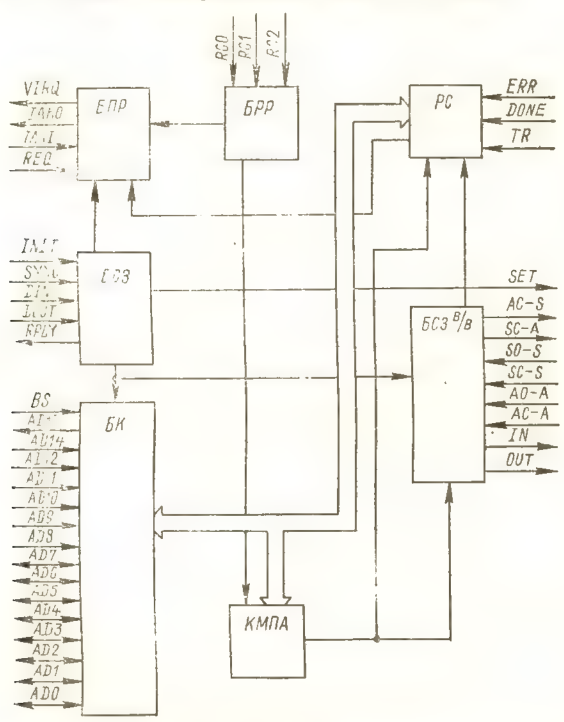


Рис. 4.19. Структурная схема микросхемы К1801ВП1-033 в режиме контроллера байтового параллельного интерфейса ввода-вывода

Расшифровка разрядов РСИ;

IEA (разряд 6) — разрешение прерывания по приему, сбрасывается канальным сигналом *INIT*;

AR (разряд 7) — требование приема;

ERR (разряд 15) — ошибка.

Расшифровка разрядов РСП:

DONE (разряд 4) — завершено;

IET (разряд 6) — разрешение прерывания по передаче, сбрасывается канальным сигналом *INIT*;

TR (разряд 7) — требование передачи;

SET (разряд 14) — начальная установка;

ERR (разряд 15) — ошибка.

При отсутствии сигнала *SO-S* в регистре *PC* установлен бит *ERR*. Это означает, что микросхема к работе не готова.

При появлении сигнала *SO-S* микросхема выставляет сигнал *AC-S*, ответом на который является сигнал *SC-S*. При поступле-

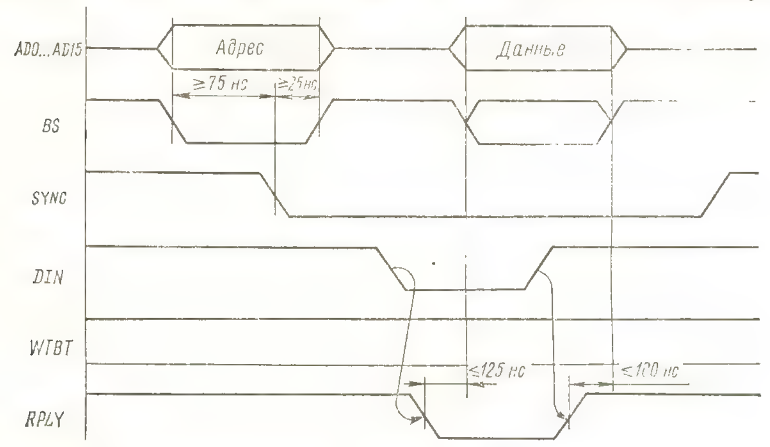


Рис. 4.20. Временная диаграмма цикла «Ввод»

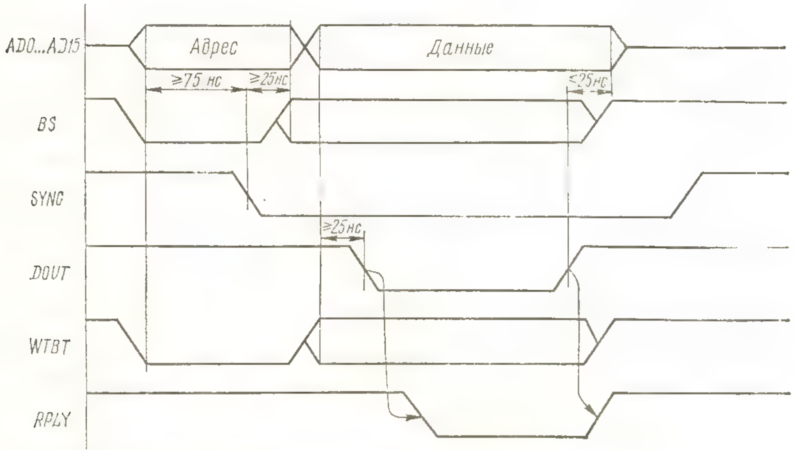


Рис. 4.21. Временная диаграмма цикла «Вывод»

нии сигнала *SC-S* в *PC* источника устанавливается бит *AR*, который при наличии бита *IEA* вызывает запрос прерывания. При чтении РП вырабатывается сигнал *IN* и снимается сигнал *AC-S*. Вновь установиться сигнал *AC-S* может только после снятия сигнала *SC-S*.

При отсутствии сигнала *AO-A* микросхема к работе не готова. При наличии сигналов *AO-A* и *AC-A* микросхема после записи

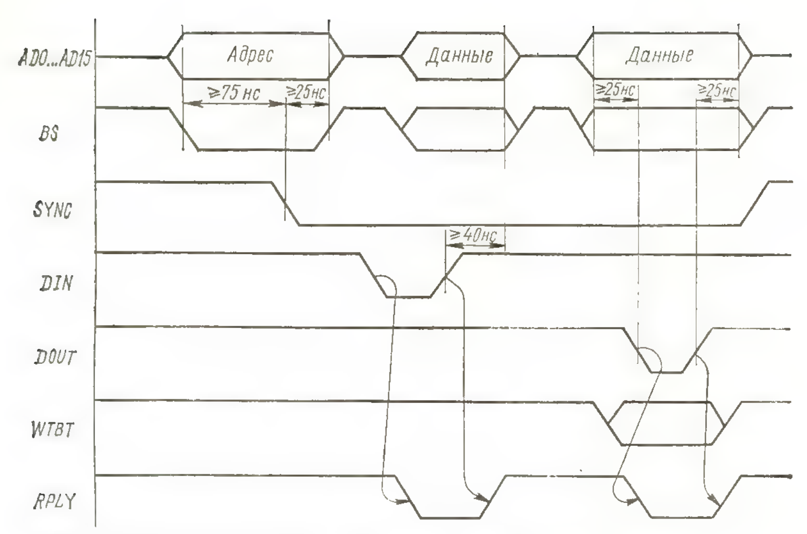


Рис. 4.22. Временна́я диаграмма цикла «Ввод—пауза—вывод»

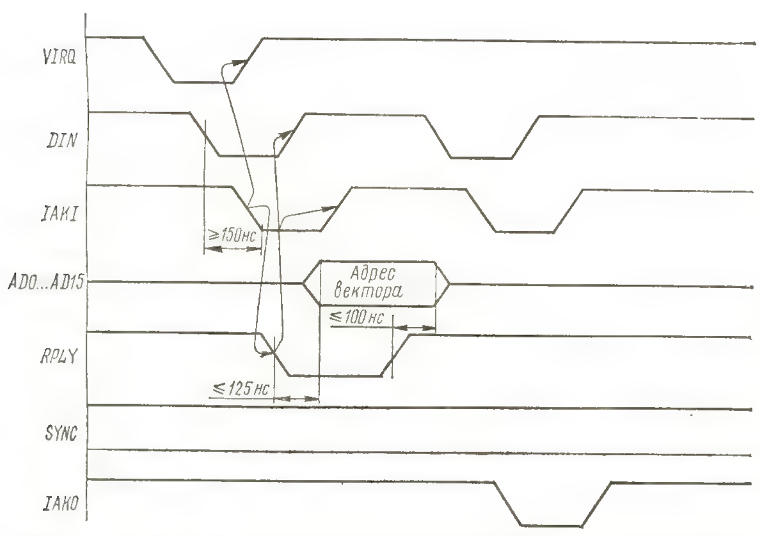


Рис. 4.23. Временна́я диаграмма выполнения процедуры прерывания

Таблица 4.18. **Статические параметры микросхемы К1801ВП1-033**

Параметр, единица измерения	Обозначение	Значение	Режим измерения
Выходное напряжение «0», В	U_{OL}	$\leqslant 0,5$	$U_{CC} = 4,75$ В $I_{OL} = 3,3$ мА
Выходное напряжение «1», В	U_{OH}	$\geqslant 2,4$	$U_{CC} = 4,75$ В $I_{OH} = 1,0$ мА
Ток утечки на входе, мкА	I_{LIH} (I_{LIL})	$\leqslant 10$	$U_{CC} = 5,25$ В $U_{IL} = 0$ В $U_{IH} = U_{CC}$
Ток потребления, мА	I_{CC}	$\leqslant 200$	$U_{CC} = 5,25$ В $U_{IL} = 0$ В $U_{IH} = U_{CC}$

информации в РИ вырабатывает сигнал *SC-A*, который снимается после снятия сигнала *AC-A*. Во время записи в РИ вырабатывается сигнал *OUT*.

Запрос на прерывание при передаче информации возникает при наличии сигналов *REQ* и установленном бите *IET* в *PC*.

Путем внешнего соединения вывода *REQ* с каким-либо выводом состояния приемника можно вызвать запрос прерывания по любому из состояний приемника. При записи сигнала «1» в 14-й разряд РСП, а также при канальном сигнале *INIT* возникает сигнал *SET*. Динамические параметры микросхемы приведены на временных диаграммах (рис. 4.20 ... 4.23), статические — в табл. 4.18.

4.6. МИКРОСХЕМА К1801ВП1-034

Эта микросхема является многофункциональным устройством и может быть использована в следующих режимах: «Устройства передачи информации»; «Устройства выдачи вектора прерывания и компаратора адреса», «Буферного регистра данных».

Условные графические обозначения микросхемы К1801ВП1-034 в указанных режимах приведены на рис. 4.24 ... 4.26.

Микросхема К1801ВП1-034 может использоваться совместно с микросхемой К1801ВП1-033 для организации устройства байтового параллельного ввода-вывода и устройства 16-разрядного программированного параллельного ввода-вывода. Выбор режима работы осуществляется подачей кода на входы *RC1* и *RC0*. В соответствии с режимом работы микросхема имеет соответствующую

структуру, условное обозначение и наименование выводов. В табл. 4.19 ... 4.21 приведены назначения выводов каждого режима в отдельности. Указанные структуры (рис. 4.27 ... 4.29) представляют собой комбинации следующих блоков: мультиплексора *М*, блока режима работы *БРР*, блока ввода-вывода *Б В/В*, буферного регистра *БР*, выходного буфера с тремя состояниями *Б Вых*, триггера *Т*, блока прерываний *БПР*, блока сравнения *БСр*.

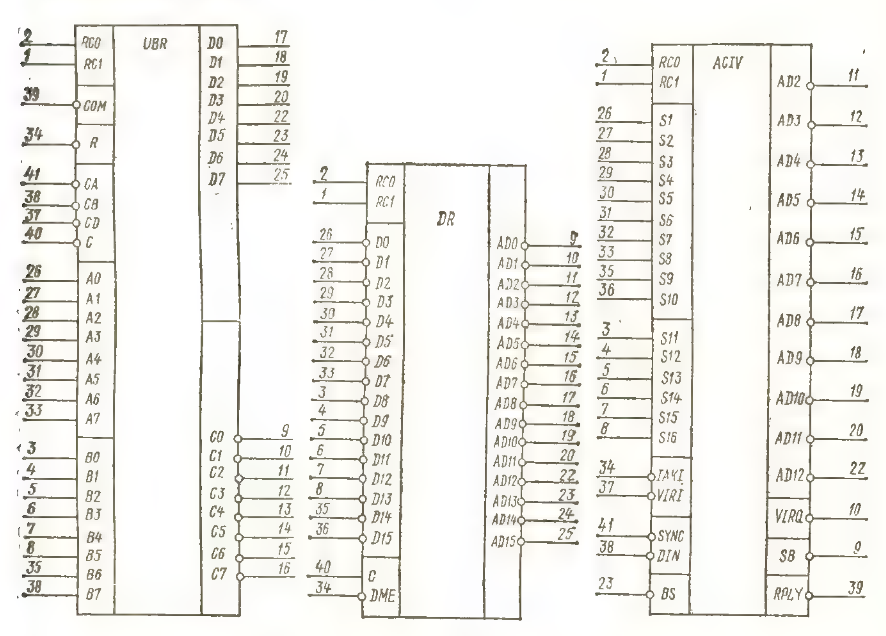


Рис. 4.24. Условное графическое обозначение микросхемы К1801ВП1-034 в режиме «Устройства передачи информации»

Рис. 4.25. Условное обозначение микросхемы К1801ВП1-034 в режиме «Буферного регистра данных»

Рис. 4.26. Условное графическое обозначение микросхемы К1801ВП1-034 в режиме «Устройства выдачи вектора прерывания и компаратора адреса»

Установка микросхемы в режим устройства передачи информации производится подачей на вывод *RC0* сигнала «1», а на вывод *RC4* — «0».

С помощью сигналов *CA* и *CB* входная информация *A0 ... A7* и *B0 ... B7* соответственно передается на двунаправленные выводы *C0 ... C7* в зависимости от сигнала *COM* в прямом или инверсном виде. Сигнал *R* обнуляет буферный регистр.

Установка микросхемы в режим буферного регистра данных производится подачей на вывод *RC0* сигнала «1», а на вывод *RC1* — «0».

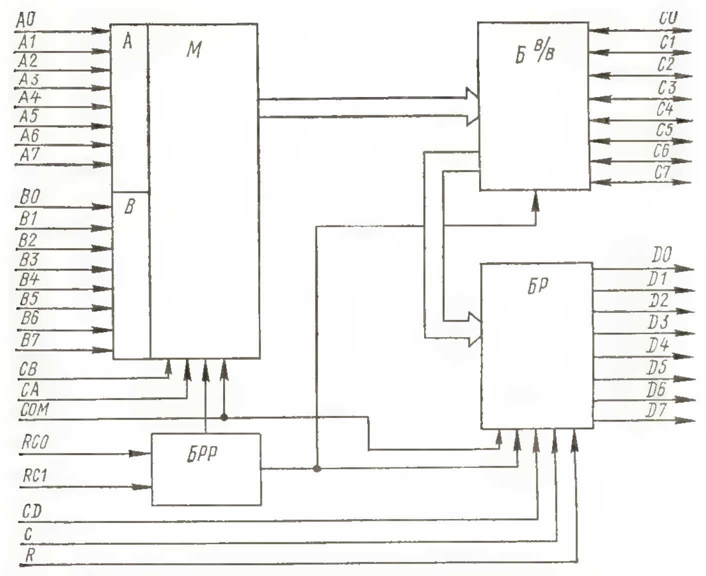


Рис. 4.27. Структурная схема микросхемы К1801ВП1-034 в режиме «Устройства передачи информации»

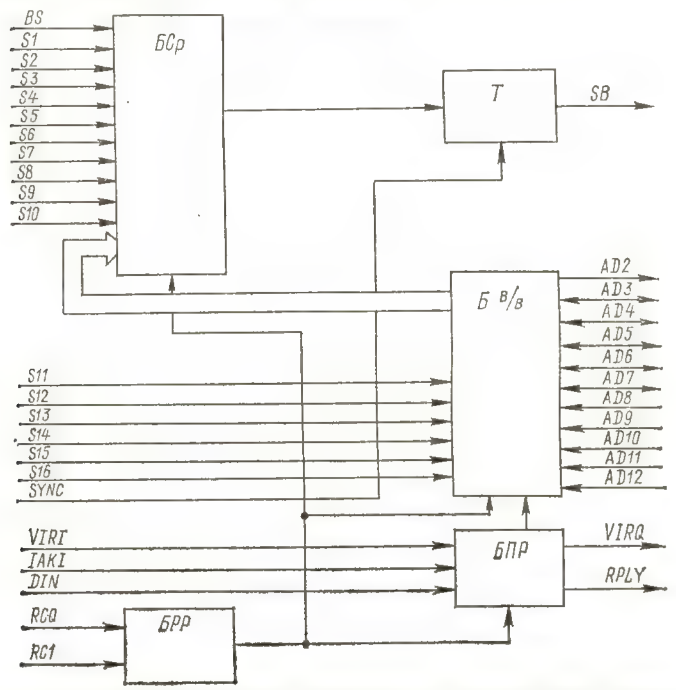


Рис. 4.28. Структурная схема микросхемы К1801ВП1-034 в режиме «Устройства выдачи вектора прерывания и компаратора адреса»

Таблица 4.19. Назначение выводов микросхемы К1801ВП1-034 в режиме устройства передачи информации

Номер вывода	Назначение	Разряд	Обозначение	Тип
1	Выбор режима	1	*RC*1	Вход
2		2	*RC*2	
3	Канал *B*	0	*B*0	
4		1	*B*1	
5		2	*B*2	
6		3	*B*3	
7		4	*B*4	
8		5	*B*5	
9	Канал *C*	0	*C*0	Вход-выход
10		1	*C*1	
11		2	*C*2	
12		3	*C*3	
13		4	*C*4	
14		5	*C*5	
15		6	*C*6	
16		7	*C*7	
17	Канал *D*	0	*D*0	Выход
18		1	*D*1	
19		2	*D*2	
20		3	*D*3	
21	Общий	—	*GND*	—
22	Канал *D*	4	*D*4	Выход
23		5	*D*5	
24		6	*D*6	
25		7	*D*7	
26	Канал *A*	0	*A*0	Вход
27		1	*A*1	
28		2	*A*2	
29		3	*A*3	
30		4	*A*4	
31		5	*A*5	
32		6	*A*6	
33		7	*A*7	
34	Сброс	—	*R*	
35	Канал *B*	6	*B*6	
36		7	*B*7	
37	Разрешение выдачи данных по каналу *D*	—	*CD*	
38	Разрешение выдачи данных по каналу *B*	—	*CB*	
39	Инвертирование	—	*COM*	
40	Запись	—	*C*	
41	Разрешение выдачи данных по каналу *A*	—	*CA*	
42	Напряжение питания	—	*Ucc*	—

Таблица 4.20. **Назначение выводов микросхемы К1801ВП1-034 в режиме буферного регистра данных**

Номер вывода	Назначение	Разряд	Обозначение	Тип
1	Выбор режима	1	*RC*1	Вход
2	Выбор режима	0	*RC*0	Вход
3	Канал *D*	8	*D*8	Вход
4	Канал *D*	9	*D*9	Вход
5	Канал *D*	10	*D*10	Вход
6	Канал *D*	11	*D*11	Вход
7	Канал *D*	12	*D*12	Вход
8	Канал *D*	13	*D*13	Вход
9	Шина адрес-данные	0	*AD*0	Выход
10	Шина адрес-данные	1	*AD*1	Выход
11	Шина адрес-данные	2	*AD*2	Выход
12	Шина адрес-данные	3	*AD*3	Выход
13	Шина адрес-данные	4	*AD*4	Выход
14	Шина адрес-данные	5	*AD*5	Выход
15	Шина адрес-данные	6	*AD*6	Выход
16	Шина адрес-данные	7	*AD*7	Выход
17	Шина адрес-данные	8	*AD*8	Выход
18	Шина адрес-данные	9	*AD*9	Выход
19	Шина адрес-данные	10	*AD*10	Выход
20	Шина адрес-данные	11	*AD*11	Выход
21	Общий	—	*GND*	—
22	Шина адрес-данные	12	*AD*12	Выход
23	Шина адрес-данные	13	*AD*13	Выход
24	Шина адрес-данные	14	*AD*14	Выход
25	Шина адрес-данные	15	*AD*15	Выход
26	Канал *D*	0	*D*0	Вход
27	Канал *D*	1	*D*1	Вход
28	Канал *D*	2	*D*2	Вход
29	Канал *D*	3	*D*3	Вход
30	Канал *D*	4	*D*4	Вход
31	Канал *D*	5	*D*5	Вход
32	Канал *D*	6	*D*6	Вход
33	Канал *D*	7	*D*7	Вход
34	Разрешение выдачи данных	—	*DME*	Вход
35	Канал *D*	14	*D*14	Вход
36	Канал *D*	15	*D*15	Вход
37	Не используются	—	—	—
38	Не используются	—	—	—
39	Не используются	—	—	—
40	Запись	—	*C*	Вход
41	Не используется	—	—	—
42	Напряжение питания	—	U_{CC}	—

Таблица 4.21. Назначение выводов микросхемы К1801ВП1-034 в режиме компаратора адреса и устройства выдачи вектора прерывания

Номер вывода	Назначение	Разряд	Обозначение	Тип
1	Выбор режима	1	*RC*1	Вход
2		0	*RC*0	
3	Вектор прерывания	11	*S*11	
4		12	*S*12	
5		13	*S*13	
6		14	*S*14	
7		15	*S*15	
8		16	*S*16	
9	Выборка устройства	—	*SB*	
10	Запрос прерывания	—	*VIRQ*	Выход
11	Шина адрес-данные	2	*AD*2	
12		3	*AD*3	Вход-выход
13		4	*AD*4	
14		5	*AD*5	
15		6	*AD*6	
16		7	*AD*7	
17		8	*AD*8	Вход
18		9	*AD*9	
19		10	*AD*10	
20		11	*AD*11	
21	Общий	—	*GND*	—
22	Шина адрес-данные	12	*AD*12	Вход
23	Внешнее устройство	—	*BS*	
24	Не используются	—	—	—
25		—	—	—
26	Адрес вектора прерывания	1	*S*1	Вход
27		2	*S*2	
28		3	*S*3	
29		4	*S*4	
30		5	*S*5	
31		6	*S*6	
32		7	*S*7	
33		8	*S*8	
34	Разрешение прерывания	—	*IAKI*	
35	Адрес вектора прерывания	9	*S*9	
36		10	*S*10	
37	Запрос прерывания	—	*VIRI*	
38	Чтение	—	*DIN*	
39	Ответ	—	*RPLY*	Выход
40	Не используется	—	—	—
41	Обмен	—	*SYNC*	Вход
42	Напряжение питания	—	*Ucc*	—

Входная информация с выводов *D0 ... D15* сигналом *C* записывается в 16-разрядный буферный регистр. Сигнал *DME* разрешает выдачу информации с буферного регистра на выводы *AD0 ... AD15*, которые при *DME*-«1» находятся в отключенном состоянии.

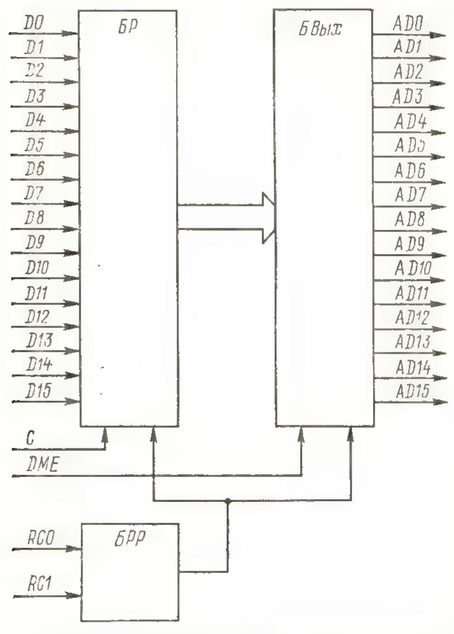


Рис. 4.29. Структурная схема микросхемы К1801ВП1-034 в режиме «Буферного регистра данных»

Установка микросхемы в режим «Устройства выдачи вектора прерывания и компаратора адреса» производится подачей на выводы *RC0* и *RC1* напряжения высокого уровня. Старшие шесть разрядов требуемого адреса вектора прерывания устанавливаются на выводах *S11 ... S16*.

Адрес, необходимый для сравнения, устанавливается на выводах *S1 ... S10*. Состояния *S1 ... S10* и *AD3 ... AD12* сравниваются при наличии сигнала *BS*. При сравнении вырабатывается сигнал *SB*, который запоминается в триггере во время присутствия сигнала *SYNC*.

Динамические параметры работы микросхемы даны на временных диаграммах (рис. 4.30 ... 4.32). Статические параметры приведены в табл. 4.22.

Таблица 4.22. **Статические параметры микросхемы К1801ВП1-034**

Параметр, единица измерения	Обозначение	Значение	Режим измерения
Выходное напряжение «0», В	U_{OL}	$\leqslant$0,5	U_{CC} = 4,75 В I_{OL} = 3,2 мА
Выходное напряжение «1», В	U_{OH}	$\geqslant$2,4	U_{CC} = 4,75 В I_{OH} = 1,0 мА
Ток утечки на входе, мкА	I_{LIL} (I_{LIH})	$\leqslant$10	U_{CC} = 5,25 В U_{IL} = 0 В U_{IH} = U_{CC}
Ток потребления, мА	I_{CC}	$\leqslant$200	U_{CC} = 5,25 В U_{IL} = 0 В U_{IH} = U_{CC}

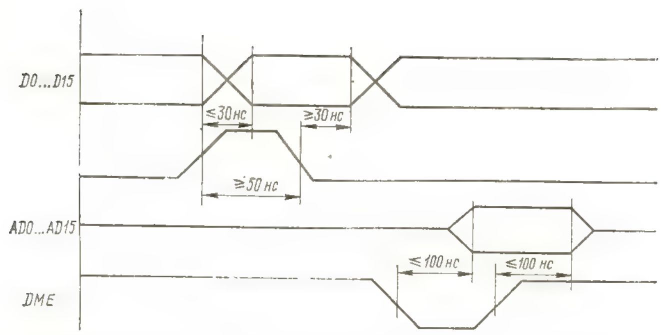


Рис. 4.30. Временна́я диаграмма работы микросхемы К1801ВП1-034 в режиме «Буферного регистра данных»

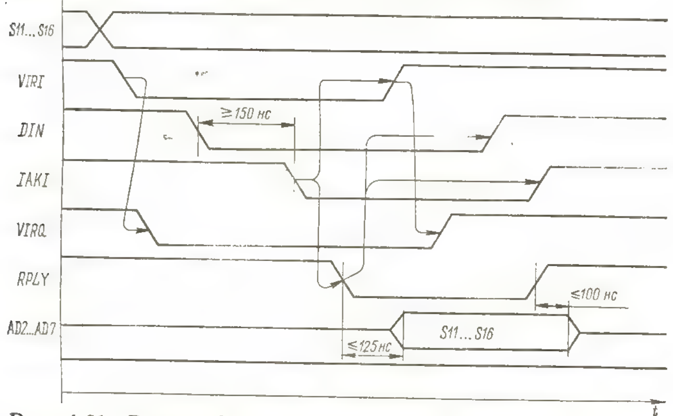


Рис. 4.31. Временна́я диаграмма работы микросхемы К1801ВП1-034 в режиме «Устройства выдачи вектора прерывания и компаратора адреса»

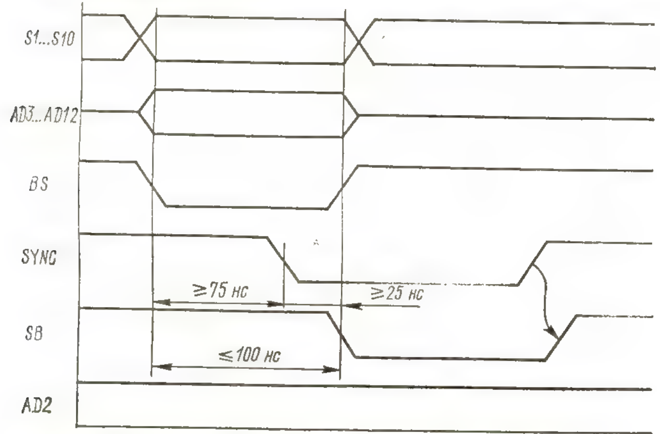


Рис. 4.32. Временна́я диаграмма работы микросхемы К1801ВП1-034 в режиме «Компаратора адреса»

Микросхема представляет синхронный приемопередатчик (БИС СПП), работающий в дуплексном режиме. БИС СПП преобразует параллельный код в последовательный и наоборот и может использоваться для связи процессора с отдаленными внешними устройствами. Микросхема имеет следующие характеристики: скорости обмена информацией (при f = 4608 кГц) — 50, 75, 100, 150, 200, 300, 600, 1200, 2400, 4800, 9600, 19200 бод; пяти-, семи- или восьмибайтовые посылки информации; формирование двух- или полуторастоповых бит; формирование битов четности или нечетности (параметра), либо работу без контроля обмена.

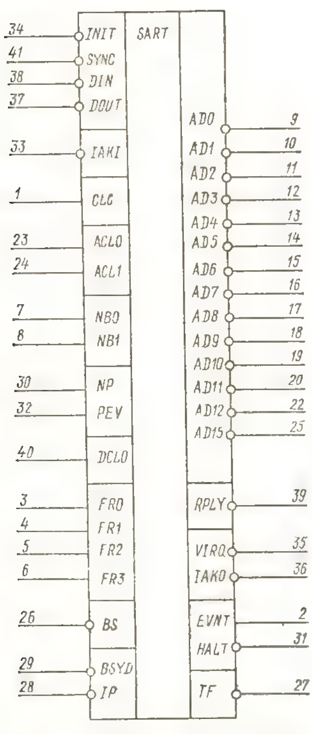


Рис. 4.33. Условное графическое обозначение микросхемы К1801ВП1-035

Условное графическое обозначение микросхемы приведено на рис. 4.33, назначение выводов — в табл. 4.23, структурная схема микросхемы — на рис. 4.34.

В состав БИС СПП входят: блок прерываний *БПР*, блок синхронизации *БСЗ*, блок системной магистрали *БСМ*, компаратор адресов и управляющих сигналов *КМПА*, блок режима работы *БРР*, блок селектора скоростей обмена *БСС*, *Приемник*, *Передатчик*.

Приемник и *Передатчик* состоят из триады регистров: буферного *БР*, сдвигового данных *СРД* и регистра состояния *РС*.

Адреса регистров и векторов прерывания приведены в табл. 4.24.

Регистр состояния приемника имеет 16-разрядную организацию. Разряды 0 ... 5, 8 ... 11, 13, 14 не используются. При чтении информация из этих разрядов воспринимается как «0».

Назначение остальных разрядов *РС* приемника заключается в следующем:

6-й — разрешение работы приемника по прерыванию. Если установлен в состояние «1», то прерывание разрешено, если в «0» — запрещено. Доступен по записи и чтению, сбрасывается в «0» по сигналу *INIT*;

7-й — флаг состояния приемника. Устанавливается в состояние «1» при поступлении посылки в *БР* приемника. Сбрасывается в «0» по окончании чтения буферного регистра приемника или по

сигналу *INIT*. Доступен по чтению. Флаг состояния приемника устанавливается в состояние «1» от предыдущей посылки или при включении источника питания;

12-й — ошибка переполнения. Устанавливается в состояние «1», если в *СРД* приемника поступило более одной посылки без чтения из *БР* приемника первой поступившей посылки. При этом независимо от числа поступивших в канал приемника посылок в *БР* сохраняется информация первой посылки. Находится в состоянии «0», если поступившая в *БР* посылка читается до окон-

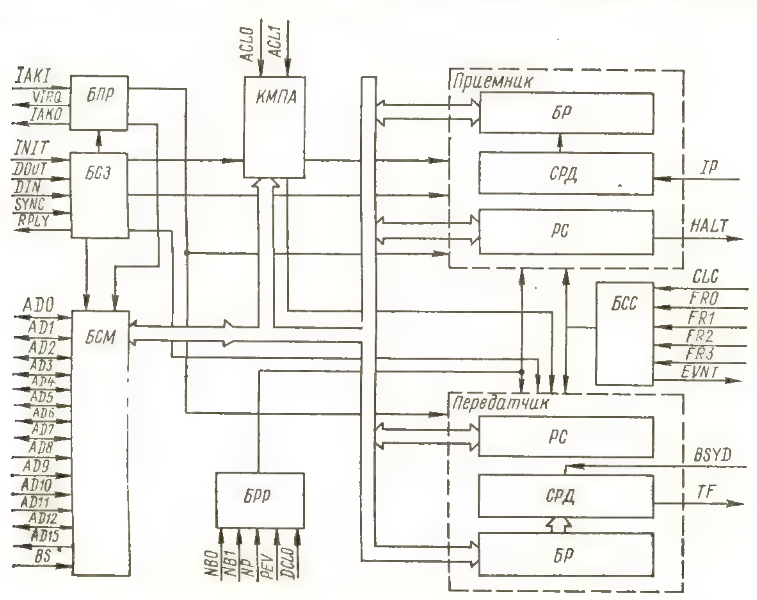


Рис. 4.34. Структурная схема микросхемы К1801ВП1-035

чания поступления в *СРД* последнего информационного бита следующей посылки. Сбрасывается в состояние «0» по окончании чтения *БР* приемника или по сигналу *INIT*. Доступен по чтению;

15-й — ошибка в принятой посылке. Устанавливается в состояние «1», если есть ошибка паритета в принятой посылке и установлено условие контроля паритета. В противном случае находится в состоянии «0». Признак ошибки сбрасывается в «0» с момента чтения содержимого *БР* приемника или по сигналу *INIT*. Доступен по чтению.

Буферный регистр приемника доступен по чтению, разряды 7 ... 0 регистра содержат посылки, принятые с линии. В нулевом разряде находится первый бит посылки, в седьмом — восьмой бит. При установленном контроле паритета в следующем за последним бите посылки находится информация о паритете. При формате «8 бит» информация анализируется микросхемой, но в *БР* не поступает.

Таблица 4.23. **Назначение выводов микросхемы К1801ВП1-035**

Номер вывода	Назначение	Разряд	Обозначение	Тип
1	Тактовая частота	—	*CLC*	Вход
2	Прерывание по таймеру	—	*EVNT*	Выход
3	Выбор скорости обмена	0	*FR*0	Вход
4		1	*FR*1	
5		2	*FR*2	
6		3	*FR*3	
7	Выбор формата	0	*NBO*	
8		1	*NB*1	
9	Шина адрес-данные	0	*AD*0	Вход-выход
10		1	*AD*1	
11		2	*AD*2	
12		3	*AD*3	
13		4	*AD*4	
14		5	*AD*5	
15		6	*AD*6	
16		7	*AD*7	
17		8	*AD*8	
18		9	*AD*9	
19		10	*AD*10	
20		11	*AD*11	
21	Общий	—	*GND*	—
22	Шина адрес-данные	12	*AD*12	Вход-выход
23	Выбор адреса	0	*ACL*0	Вход
24		1	*ACL*1	
25	Шина адрес-данные	15	*AD*15	Вход-выход
26	Выбор внешнего устройства	—	*BS*	Вход
27	Передатчик	—	*TF*	Выход
28	Приемник	—	*IP*	Вход
29	Последовательный канал занят	—	*BSYD*	
30	Установка паритета	—	*NP*	
31	Останов	—	*HALT*	Выход
32	Четность-нечетность	—	*PEV*	Вход
33	Разрешение прерывания	—	*IAKI*	
34	Установка	—	*INIT*	
35	Запрос прерывания	—	*VIRQ*	Выход
36	Разрешение прерывания	—	*IAKO*	
37	Вывод	—	*DOUT*	Вход
38	Ввод	—	*DIN*	
39	Ответ	—	*RPLY*	Выход
40	Авария источника питания	—	*DCLO*	Вход
41	Обмен	—	*SYNC*	
42	Напряжение питания	—	*Ucc*	—

Таблица 4.24. **Адресация регистров и векторов прерывания микросхемы К1801ВП1-035**

Тип регистра и адрес вектора прерывания	Группы			
	1	2	3	4
Регистр состояния приемника	177560	176560	176570	XXXXX0
Буферный регистр приемника	177562	176562	176572	XXXXX2
Регистр состояния передатчика	177564	176564	176574	XXXXX4
Буферный регистр передатчика	177566	176566	176576	XXXXX6
Адрес вектора прерывания приемника	060	360	370	XX0
Адрес вектора прерывания передатчика	064	364	374	XX4

Низкий уровень бита в посылке на входе *IP* соответствует сигналу «1», высокий уровень — «0» в *БР* приемника. Разряды 8 ... 15 не используются, их информация воспринимается как «0». Регистр состояния передатчика имеет 16-разрядную организацию. Назначение разрядов следующее:

нулевой — разрыв линии. Если установлен в «1», то на выходе *TF* устанавливается «1» (состояние «Старт») при условии, что на входе *BSYD* — «0». Если на входе *BSYD* — «1», то на выходе *TF* устанавливается «0» (состояние «Стоп»). В этом случае посылка из микросхемы не выходит и установка нулевого разряда в «1» не изменяет сигнала на выходе *TF*. Доступен по записи и чтению. Сбрасывается в состояние «0» по сигналу *INIT*;

2-й — проверка работы. Если установлен в состояние «1», то выдаваемая с выхода *TF* посылка поступает также и в канал приемника. Разряд сбрасывается в состояние «0» по сигналу *INIT*. В этом случае вход *IP* для приема посылок с линии закрыт. Доступен по записи и чтению;

6-й — разрешение работы передатчика по прерыванию. Если установлен в состояние «1», то прерывание разрешено, если в «0» — запрещено. Доступен по записи и чтению, сбрасывается в состояние «0» по сигналу *INIT*;

7-й — флаг состояния передатчика. Устанавливается в состояние «1» в момент выдачи посылки на линию или по сигналу *DCLO*. Сбрасывается в состояние «0» по записи информации в *БР* передатчика. Свидетельствует о том, что *БР* передатчика пуст. Если запись информации в *БР* передатчика произошла во время выдачи посылки, то следующая посылка начинается сразу же по окончании предыдущей. Доступен по чтению.

Разряды 1, 3 ... 5 и 8 ... 15 не используются. Их информация воспринимается как «0».

Буферный регистр передатчика имеет 16-разрядную организацию. Доступен по записи.

Разряды 0 ... 7 являются данными для передачи посылки на линию. В нулевой разряд записывается первый бит посылки, в 7-й—8-й бит. По окончании записи в *БР* передатчика происходит запись данных в *СРД*. С выхода *TF* посылка поступает на линию при условии, что на входе *BSYD* «0».

При соблюдении условия формирования паритета этот бит автоматически следует за последним битом посылки.

При записи «1» в буферный регистр *передатчика* данный бит поступает с выхода *TF* в виде «0», при записи «0» — «1».

При чтении по адресу буферного регистра передатчика читается адрес источника вектора прерывания. В этом случае разряды 8 ... 15 выдают «0».

Прерывание от приемника возникает, если 6-й и 7-й разряды *PC* приемника установлены в «1». При обработке запроса на прерывание происходит считывание адреса вектора прерывания приемника. Запрос на прерывание от передатчика возникает, если 6-й и 7-й разряды *PC* передатчика установлены в «1». При обработке запроса на прерывание читается адрес вектора прерывания передатчика.

Формат источников адресов векторов прерываний представляет 16-разрядную структуру, в которой 2-й разряд — указатель адреса вектора прерывания приемника; «1» в этом разряде указывает на адрес передатчика; разряды 3 ... 7 — сменная часть адреса вектора прерывания. Если выбрана четвертая группа адресов регистров и источников адресов векторов прерываний, то разряды 3 ... 7 считываются как «0». Разряды 0, 1, 8 ... 15 не используются и также считываются как «0».

Следует отметить, что приоритет запроса от приемника выше приоритета запроса от передатчика.

По окончании приема посылки микросхема устанавливает на выходе *HALT* «0», «1» образуется по сигналу *INIT*.

При тактовой частоте 4608 кГц на входе *EVNT* микросхема обеспечивает частоту импульсных сигналов 50 Гц со скважностью 2.

Выбор групп адресов регистров и источников адресов векторов прерываний производится заданием комбинаций уровней на входах *ACL0* и *ACL1* (см. табл. 4.25).

При комбинации *ACL0* = «1», *ACL1* = «1» регистры микросхемы выбираются по адресу, состоящему из 1-го и 2-го разрядов, и по сигналу *BS*. На разряды 12 ... 3 адреса микросхема не реагирует. Группа 4 может быть использована для внешнего произвольного набора адресов регистров и внешнего источника адресов векторов прерывания. Выбор форматов посылок по последовательному каналу производится заданием комбинаций уровней на входах *NB0* и *NB1* (табл. 4.26).

Выбор скорости обмена по последовательному каналу производится заданием комбинаций уровней на входах *FR0* ... *FR3* (табл. 4.27).

Таблица 4.25. **Адресация групп регистров и источников прерываний**

Обозначение	Группа			
	1	2	3	4
*ACL*0	0	1	0	1
*ACL*1	0	0	1	1

Таблица 4.26. **Форматы последовательных посылок**

Обозначение	Длина посылки, бит		
	6	7	8
*NB*0	X	0	1
*NB*1	0	1	1

Таблица 4.27. **Управление скоростью обмена информацией**

Скорость обмена, бод	*FR*3	*FR*2	*FR*1	*FR*0
50	0	0	0	0
75	0	0	0	1
100	0	0	1	0
150	0	0	1	1
200	0	1	0	0
300	0	1	0	1
600	0	1	1	0
1 200	0	1	1	1
2 400	1	0	0	0
4 800	1	0	0	1
9 600	1	0	1	0
19 200	1	0	1	1
57 600	1	1	0	0

Таблица 4.28. **Управление режимом работы**

Режим работы с паритетом	*NP*	*PEV*
Нет формирования бита паритета и контроля паритета	1	X
Формирование бита четности и контроль по четности	0	1
Формирование бита нечетности и контроль по нечетности	0	0

Таблица 4.29. **Режимы работы передатчика**

Режим	Вход *BSYD*	Нулевой разряд PC	Выход *TF* передатчика
Выдача посылки, линия свободна	0	0	Выдача посылки
Линия занята	1	X	0 (Стоп)
Разрыв линии	0	1	1 (Старт)
Подача сигнала *DCL*0	X	X	0 (Стоп)

Выбор режима с паритетом или без него, а также режима работы с паритетом по четности или нечетности задается комбинациями уровней на входах *NP* и *PEV* (табл. 4.28).

В табл. 4.29 приведены режимы работы передатчика.

Микросхема имеет выводы *HALT*, *VIRQ* и *RPLY* с открытым стоком, обеспечивающим только «0». Для «1» необходимо между выходом и шиной источника питания включать резистор с номиналом, рассчитанным исходя из значения выходного тока «0» (I_{OL} не более 3,2 мА).

Для работы микросхемы со специальными последовательными линиями типа линий с токовыми петлями, телеграфным интерфейсом и другими необходимы специальные схемы сопряжения (шинные формирователи, магистральные усилители и т. д.). Начальная установка микросхемы производится подачей сигналов «0» на входы *INIT* и *DCLO*.

Длительность сигналов на входе *INIT* — не менее 10 мкс, на входе *DCLO* — не менее 5 мс.

Статические параметры микросхемы приведены в табл. 4.30.

Таблица 4.30. **Статические параметры микросхемы К1801ВП1-035**

Параметр, единица измерения	Обозначение	Значение	Режим измерения
Входное напряжение «0», В	U_{OL}	$\leqslant 0{,}5$	$U_{CC} = 4{,}75$ В $I_{OL} = 3{,}2$ мА
Выходное напряжение «1», В	U_{OH}	$\geqslant 2{,}5$	$U_{CC} = 4{,}75$ В $I_{OH} = 1{,}0$ мА
Ток утечки на входе, мкА	I_{LIL} (I_{LIH})	$\leqslant 10$	$U_{CC} = 5{,}25$ В $U_{IL} = 0$ В $U_{IH} = U_{CC}$
Ток потребления, мА	I_{CC}	$\leqslant 200$	$U_{CC} = 5{,}25$ В $U_{IL} = 0$ В $U_{IH} = U_{CC}$

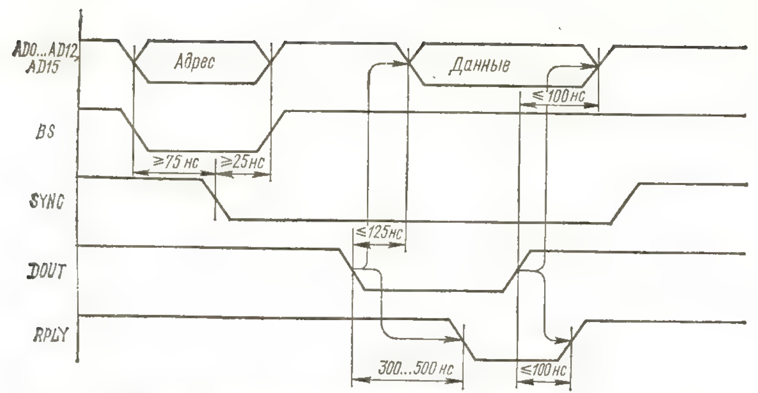


Рис. 4.35. Временна́я диаграмма режима «Чтение»

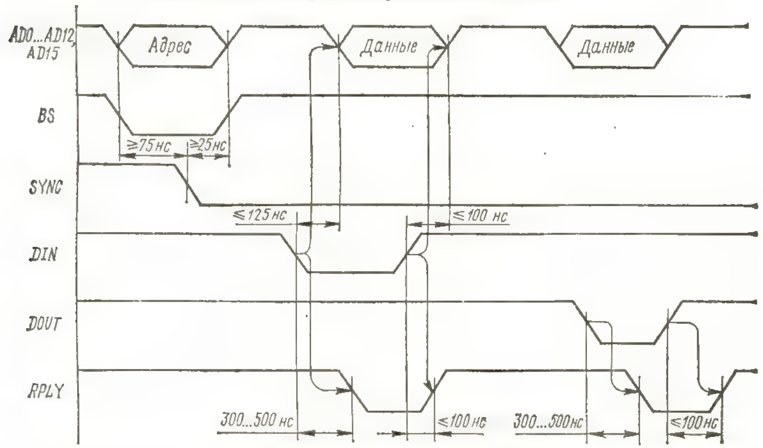


Рис. 4.36. Временна́я диаграмма режима «Чтение—пауза—запись»

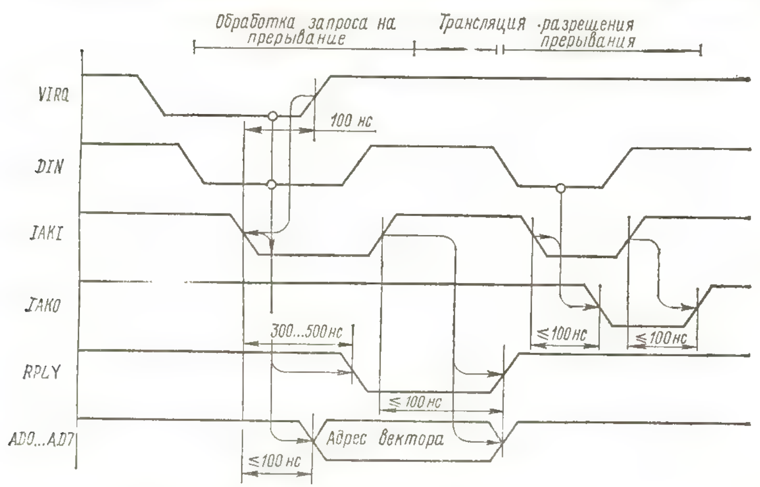


Рис. 4.37. Временна́я диаграмма режима «Прерывание»

На временных диаграммах (рис. 4.35 ... 4.37) приведены динамические параметры микросхемы К1801ВП1-035 в различных режимах работы, а также иллюстрируется работа микросхемы при выполнении различных операций обмена.

4.8. ПРИМЕР ПРИМЕНЕНИЯ МИКРОСХЕМ СЕРИИ К1801

В качестве примера применения рассмотрим структуру микроЭВМ «Электроника НМС11100.1». Структурная схема микроЭВМ представлена на рис. 4.38 [50, 55, 59].

МикроЭВМ состоит из следующих основных устройств: микропроцессор, оперативное запоминающее устройство *ОЗУ*, системное постоянное запоминающее устройство *СПЗУ*, устройство байтового параллельного обмена информацией *УБПИ*, устройство последовательного ввода-вывода данных *УПВД*, интерфейс накопителя на гибких магнитных дисках *НГМД*, контактное устройство для ПЗУ пользователя *КУ ПЗУ*, корректор сигналов канала *КСК*, схема начального пуска *СНП*, блок генераторов импульсов *ГИ*, схемы преобразования напряжения питания *ПН1* и *ПН2*, схемы организации линий связи *ЛС1* ... *ЛС4*.

МикроЭВМ «Электроника НМС11100.1» (далее ЭВМ) аналогична по конструктивному исполнению, системе команд и интерфейсу микроЭВМ «Электроника-60». Архитектурные особенности ЭВМ позволяют подключать через канал обмена дополнительные типовые узлы и периферийное оборудование микроЭВМ «Электроника-60», а также использовать ее программное обеспечение.

Рассмотрим подробнее структуру ЭВМ.

Процессор *ПРЦ* является основным устройством ЭВМ. Он выполняет необходимые операции по приему и выполнению команд, обработке прерываний и по управлению каналом обмена. *ПРЦ* выполнен на основе микросхемы К1801ВМ1. Обрабатывает 16-разрядную информацию (данные или адрес) в соответствии с командной информацией и сигналами управления. Работа *ПРЦ* синхронизируется синхросигналами с блока *ГИ*. Обмен информацией *ПРЦ* с каналом происходит через *КСК*, представляющий блок усилителей, построенных на базе микросхем К531АП2П.

Усилители шины адрес-данные работают в двунаправленном режиме, усилители сигналов управления каналом и процессором — в однонаправленном.

Устройство *ОЗУ* предназначено для временного хранения информации ЭВМ. *ОЗУ* содержит: накопитель *НОЗУ*, устройство управления *УОЗУ*, буферный регистр данных *БРД*, блок управления банками *БВБ*.

НОЗУ выполнен на базе микросхем К565РУ3А и имеет организацию 32К × 16 разрядов. Объем памяти разделен на 8 банков по 4К каждый: *УОЗУ* обеспечивает обмен информацией между *ОЗУ* и каналом ЭВМ. Основой устройства является микросхема К1801ВП1-030. *БРД* предназначен для временного хранения дан-

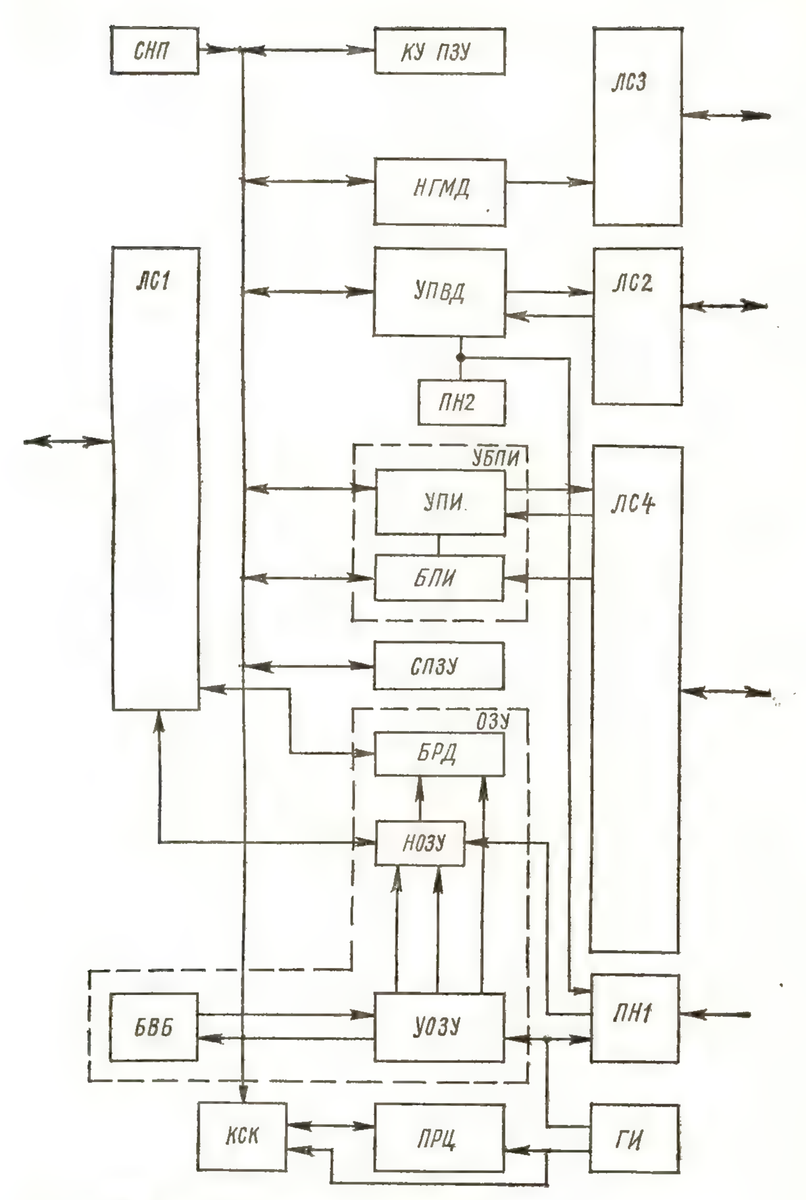


Рис. 4.38. Структурная схема микроЭВМ

ных после завершения их выборки из *ОЗУ*. Данные в *БРД* не изменяются в течение всего цикла обмена *ОЗУ* с каналом ЭВМ.

БВБ предназначен для отключения отдельных банков *ОЗУ* из области адресации ЭВМ. Отключение производится внешними движковыми переключателями. Блок выполнен на базе микросхем К155КП7.

Устройство *УБПИ* предназначено для связи ЭВМ с внешними устройствами ВУ по асинхронным параллельным каналам. Это устройство содержит: блок управления интерфейсом *УПИ*, выполненный на базе микросхемы К1801ВП1-033, блок передачи *БПИ*, выполненный на базе микросхемы К1801ВП1-034.

Устройство *УПВД* предназначено для связи ЭВМ с ВУ по последовательному асинхронному каналу.

Выбор режимов работы, скоростей обмена, а также адресов векторов прерывания осуществляется внешними переключателями, связь с внешними устройствами — узлом оптронной развязки по линии типа «20 мА токовая петля».

Итерфейс *НГМД* предназначен для организации обмена ПРЦ с накопителями на гибких магнитных дисках. Обмен происходит через регистры команд и данных. Адресация регистров осуществляется переключателями. Контактное устройство *КУ ПЗУ* предназначено для установки в него микросхемы ПЗУ типа К1801РЕ1 с жесткой «прошивкой».

В адресном пространстве ПЗУ может быть установлено вместо любого, отключенного банка *ОЗУ*. Схема *СНП* предназначена для генерации начального адреса программы запуска ЭВМ. Блок *ГИ* выполняет стандартную функцию формирования тактовых импульсов (ТИ). Частоты следования импульсов составляют 8; 40; 4608 кГц.

Схемы *ПН1*, *ПН2* предназначены для формирования питающего напряжения (—5 В) для *НОЗУ*. Схемы *ПН1*, *ПН2* выполнены по принципу «удвоения напряжения».

Схемы организации линии связи предназначены для обеспечения нормальной работы ЭВМ.

В заключение следует сказать, что описанная микроЭВМ является управляющим ядром диалоговых вычислительных комплексов (ДВК). ДВК используют в качестве рабочего места разработчика в системах автоматизации проектирования. МикроЭВМ «Электроника НМС11100.1» может работать совместно с внешними системными ПЗУ, а также с алфавитно-цифровыми дисплеями, термопечатающим устройством и накопителем на гибких магнитных дисках.

МикроЭВМ удобно использовать для оперативного выполнения навигационных, аварийных и грузовых расчетов. К ним, в частности, относятся и определение расстояния до конечной точки, автосчисление пути судна по заданному расстоянию, расчет координат судна по данным приемоиндикаторов, определение места судна и поправки компаса по Солнцу и звездам [8, 48].

Глава 5

МИКРОПРОЦЕССОРНЫЙ КОМПЛЕКТ БИС СЕРИИ К1810 (МИКРОСХЕМА К1810ВМ86)

5.1. ОБЩИЕ СВЕДЕНИЯ

Широкое распространение микросхем серии КР580 стимулировало не только ее дальнейшее развитие, но и создание нового процессорного устройства, обладающего свойствами 8- и 16-разрядных МП, — микросхемы К1810ВМ86. При проектировании микросхемы К1810ВМ86 ставилась цель пропорционально улучшить существующие характеристики микросхемы КР580ИК80 и расширить ее возможности. Дополнительные возможности — это 16-разрядные арифметические операции, операции над байтовыми последовательностями с возможностью обработки битов информации, повторная входимость программ, независимость программ от размещения в ЗУ, динамическое перемещение модулей, прямая адресация к ЗУ емкостью до 1 Мбайта [36 ... 38].

Микросхема К1810ВМ86 способна реализовать высокопроизводительные комплексы, применяемые, например, в системе организации управления судном из одного центра, а также для решения задач гидролокации. Повышение производительности судовых систем управления требует использования памяти значительного объема, а также организации управления разветвленной сетью медленно действующих внешних контроллеров. Именно поэтому разработчикам судовой РЭА следует обратить внимание на возможности микросхемы К1810ВМ86 [8].

Микросхема К1810ВМ86 представляет однокристальный 16-разрядный МП, выполненный по *n*-МОП-технологии. Микропроцессор К1810ВМ86 (МП) обладает возможностью адресации к памяти емкостью до 1 Мбайта, выполняет операции над битами, байтами и 16-разрядными словами, выполняет команды умножения и деления, двоичной и десятичной арифметики. МП, так же как и микросхему КР580ИК80, нельзя наращивать и микропрограммировать. В МП возможно до 256 типов прерываний, он совмещает операции выполнения текущей команды и выборки следующей, а также работает в минимальном и максимальном режимах. Это позволяет расширить области применения МП, начиная с простых контроллеров и кончая мультипроцессорными системами. В последнем случае координируется взаимодействие нескольких процессоров (сопроцессоров).

Микросхема может работать на одну ТТЛ-нагрузку; период следования тактовых синхроимпульсов от 200 до 500 нс, мощность потребления микросхемой не более 1,75 Вт.

Таблица 5.1. **Назначение выводов микросхемы К1810ВМ86**

Номер	Назначение	Разряд	Обозначение	Тип
1	Общий	—	*GND*	—
2		0	*AD*0	
3		1	*AD*1	
4		2	*AD*2	
5		3	*AD*3	
6		4	*AD*4	
7		5	*AD*5	
8		6	*AD*6	
9	Шина адрес-данные	7	*AD*7	Вход-выход
10		8	*AD*8	
11		9	*AD*9	
12		10	*AD*10	
13		11	*AD*11	
14		12	*AD*12	
15		13	*AD*13	
16		14	*AD*14	
17	Немаскируемый запрос прерывания	—	*NMI*	
18	Запрос прерывания	—	*INT*	Вход
19	Синхронизация	—	*CLC*	
20	Общий	—	*GND*	—
21	Установка	—	*CLR*	
22	Готовность	—	*RDY*	Вход
23	Проверка состояния	—	*TEST*	
24 ... 31	См. табл. 5.2 и 5.3	—	—	—
32	Чтение	—	*R*	Выход
33	Режим работы	—	*NM/MX*	Вход
34	Разрешение передачи старшего байта либо 7-го разряда состояния	— / 7	BHE / *ST*7	
35		19 / 6	*A*19 / *ST*6	
36	Шина адрес-состояние: в такте Т1 выдается адрес, в тактах Т2 ... Т4 — состояние	18 / 5	*A*18 / *ST*5	Выход
37		17 / 4	*A*17 / *ST*4	
38		16 / 3	*A*16 / *ST*3	
39	Шина адрес-данные	15	*AD*15	Вход-выход
40	Напряжение питания	—	U_{cc}	—

Примечание. Выводы 2 ... 16, 32, 34 ... 39 трехстабильные, к выводу 33 возможно подключение напряжения питания.

Таблица 5.2. **Назначение выводов 24 ... 31 микросхемы К1810ВМ86 в максимальном режиме**

Номер вывода	Назначение	Разряд	Обозначение	Тип
24	Код состояния очереди команд	0	*QS*0	Выход
25		1	*QS*1	
26	Состояние цикла шины, изменение разряда в такте Т4 указывает на начало цикла	2	*ST*2	
27		1	*ST*1	
28		0	*ST*0	
29	Шина занята	—	*LOCK*	
30	Запрос-разрешение доступа к шине адрес-данные; имеет более высокий приоритет, чем *RQ/EI*, в тактах Т4 или Т5 МП выдает сигнал разрешение доступа	—	*RQ/EO*	Вход-выход
31	Запрос-разрешение доступа к шине адрес-данные	—	*RQ/EI*	

Примечание. Выводы 26 ... 29 трехстабильные.

В табл. 5.1 приведено назначение выводов МП, в табл. 5.2 и 5.3 — назначение выводов в двух режимах работы. Структурная схема МП представлена на рис. 5.1.

В состав структурной схемы входят следующие узлы и блоки: схема обработки запросов прерывания *СЗП*, схема внутренней синхронизации *СВС*, схема управления доступом к шине *СУД*, регистры очереди команд *РгОК*, микропрограммное устройство

Таблица 5.3. **Назначение выводов 24 ... 31 микросхемы К1810ВМ86 в минимальном режиме**

Номер вывода	Назначение	Разряд	Обозначение	Тип
24	Подтверждение прерывания	—	*INTA*	Выход
25	Строб адреса (в такте Т1)	—	*STB*	
26	Разрешение передачи данных (выдается в каждом такте)	—	*DE*	
27	Выдача-прием данных	—	*OP/IP*	
28	Доступ к ЗУ или устройству ввода-вывода	—	*M/IO*	
29	Запись (выдается в тактах Т2, Т3 и тактах ожидания)	—	*W*	
30	Подтверждение захвата	—	*HLDA*	Вход
31	Запрос захвата	—	*HLD*	

Примечание. Выводы 26 ... 29 трехстабильные.

управления *МУУ*, сумматор адреса *СмА*, сегментные регистры *СР*, 16-разрядное арифметико-логическое устройство *АЛУ*, регистры общего назначения *РОН*, буферы шины адрес-данные *БАД*, схема управления циклами обмена *СУЦО*.

Указанные узлы и блоки объединяются в два крупных устройства: устройство обработки *УО* и устройство сопряжения с шиной *УСШ*. *УО* предназначено для декодирования и выполнения команд и состоит из *АЛУ*, *РОН*, *МУУ*, *СЗП*, *СУД*.

АЛУ выполнено на базе 16-разрядного комбинационного сумматора с последовательно-параллельным переносом. Оно включает три регистра: регистры временного хранения операндов и

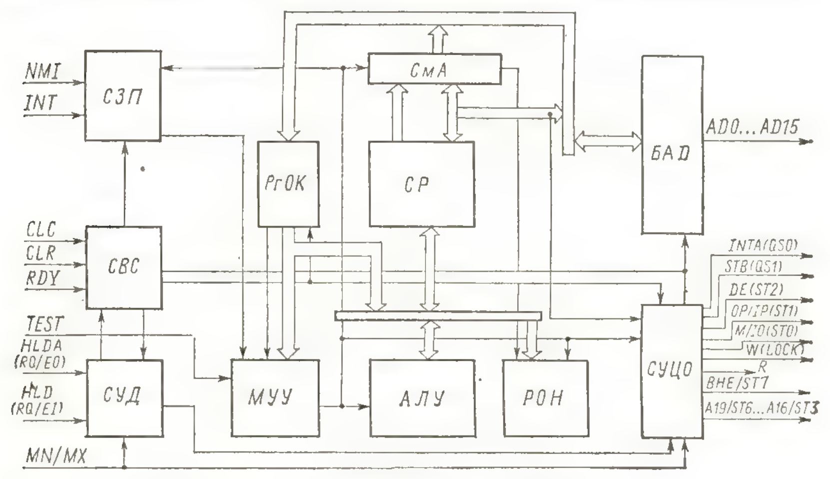


Рис. 5.1. Структурная схема микросхемы К1810ВМ86

результатов и регистр признаков *РП*. Обмен данными осуществляется через *РОН* (восемь 16-разрядных регистров). *МУУ* предназначено для управления выполнением команд путем декодирования команды и выработки необходимых управляющих сигналов. *СЗП* управляет структурой прерываний как от внешних источников (сигналы *NMI* и *INT*), так и от внутренних. *СУД* служит для управления возможностью предоставления шины адрес-данные внешним устройствам, способным организовывать режим прямого доступа к памяти.

Устройство обработки *УО* осуществляет связь с внешними (по отношению к МП) микросхемами через *УСШ*. В состав УСШ входят схемы: *СмА*, *СР*, *СВС*, *СУЦО*, *РгОК*.

Сумматор адреса *СмА* совместно с 16-разрядным регистром адреса команды *РАК*, входящим в *СР*, обеспечивает хранение смещения команды от начала сегмента кода и вместе с тем указывает следующую команду. Базовый адрес сегмента кода и смещение образуют логический адрес команды. При любых обращениях к ЗУ происходит преобразование логического адреса в физиче-

ский, который получается путем сдвига базового адреса сегмента на четыре разряда и сложения его с величиной смещения. Сложение происходит по модулю 2^{16}. В этой процедуре заняты внутренние регистры сегментов кода данных, относящиеся к *СР*.

Регистры *РгОК* предназначены для реализации очереди команд. Они, по существу, относятся как к *УО*, так и к *УСШ*.

В *РгОК* можно хранить до 6 байтов кода команд. Схема *СВС* служит для преобразования поступающих извне тактовых импульсов в две неперекрывающиеся последовательности, управляющие работой узлов и устройств МП. Кроме того, *СВС* обеспечивает синхронизацию обмена информацией МП с медленными устройствами.

Рассмотрим подробнее наиболее важные внутренние узлы МП.

РОН МП включают восемь 16-разрядных регистров, разделенных на две группы — регистры данных и индексные регистры. Регистры данных можно использовать как 16-, так и 8-разрядные с отдельной адресацией. Эти регистры можно задавать явно и

Таблица 5.4. **Задание регистров данных**

Регистр (разряды)	Обозначение регистра	Назначение
Аккумулятор (0 ... 15)	*AX*	Умножение, деление, ввод-вывод слов
Аккумулятор (0 ... 7)	*AL*	Умножение, деление, ввод-вывод байтов, перекодировка, десятичная арифметика
Аккумулятор (8 ... 15)	*AH*	Умножение, деление байтов
База (0 ... 15)	*BX*	Перекодировка
Счетчик (0 ... 15)	*CX*	Операция со строками, циклы
Счетчик (0 ... 7)	*CL*	Сдвиги и циклические сдвиги
Регистр данных (0 ... 15)	*DX*	Умножение, деление слов, неявный ввод-вывод
Указатель стека (0 ... 15)	*SP*	Операции со стеком
Указатель базы (0 ... 15)	*BP*	Операции адресации
Индекс источника (0 ... 15)	*SI*	Операции со строками
Индекс приемника (0 ... 15)	*DI*	То же

Таблица 5.5. Назначение разрядов РП

Номер разряда	Обозначение	Назначение
0	*CF*	Признак переноса
2	*PF*	Признак четности
4	*CnF*	Признак вспомогательного переноса
6	*ZF*	Признак нуля
7	*SF*	Знак результата
8	*TF*	Признак пошагового режима
9	*IF*	Признак разрешения прерывания
10	*DF*	Признак направления
11	*OF*	Признак переполнения

неявно (табл. 5.4). Индексные регистры включают регистр сегмента кода *CS*, регистр сегмента данных *DS*, регистр сегмента стека *SS*, дополнительный сегмент *ES*, регистр адреса команд *IP*.

Все РОН подходят под определение «аккумулятор», используемое при описании микросхемы КР580ИК80 (серия КР580).

Выход АЛУ связан с РП. В РП значащими являются девять разрядов (табл. 5.5).

В системе команд МП имеется целая группа команд условных переходов, которые используют признаки результата. Признаки *CF*, *ZF*, *SF*, *OF*, *CnF* формируются стандартным образом (см., например, описание микросхемы КР580ИК80). Если признак четности *PF* = «1», то результат операции четный. Признак используется для осуществления контроля по четности передач информации по шине адрес-данные.

Если признак направления *DF* = «1», то это вызывает автодекремент при выполнении операций со строками данных. Другими словами, строки в этом случае обрабатываются справа налево, от старших адресов к младшим. Логический нуль в этом разряде РП вызывает автоинкремент, т. е. обработку строк слева направо.

Если признак разрешения прерывания *IF* = «1», то МП реагирует на внешние запросы прерывания. Если признак пошагового режима *TF* установлен в «1», то МП после выполнения каждой команды автоматически генерирует прерывание.

МП может обрабатывать до 256 типов прерываний, которые делят на три группы: внешние, внутренние и программные.

Внешние прерывания поступают на выводы *INT* и *NMI*. По входу *INT* поступают маскируемые запросы, которые маскируются до окончания текущей команды и только при взведении внутреннего признака разрешения прерывания удовлетворяются. По входу *NMI* поступают немаскируемые запросы, которые удовлетворяются в конце текущей команды независимо от состояния признака разрешения прерывания.

К внутренним относятся немаскируемые прерывания по переполнению, по ошибке деления и при пошаговом режиме.

Программные прерывания реализуются после выполнения команды разрешения прерывания INT. Тип прерывания закодирован в самой команде.

Несколько слов следует сказать о сегментных регистрах. Вся адресуемая МП память (до 1 Мбайта) разделена на логические сегменты по 64 Кбайт каждый. МП имеет одновременный доступ к четырем сегментам благодаря наличию сегментных регистров. В последних хранятся базовые (начальные) адреса сегментов, а именно:

CS содержит базовый адрес сегмента, из которого выбираются команды;

DS и *SC* указывают текущий сегмент данных и стека соответственно. Регистр сегмента стека *SC* указывает текущий сегмент стека. С ячейками *SC* производят стековые операции;

ES определяет начало сегмента, используемого для хранения данных.

5.2. ФУНКЦИОНИРОВАНИЕ МИКРОСХЕМЫ К1810ВМ86

Для запуска микросхемы одновременно с напряжением питания подается сигнал *CLR* высокого уровня, который должен оставаться в состоянии высокого уровня не менее 50 мкс после достижения номинального значения напряжения питания. Для установки МП в исходное состояние необходимо подавать на вход *CLR* сигнал высокого уровня в течение не менее четырех периодов тактовой частоты. Вслед за этим сигналом МП прекращает выполнение внутренних операций и переводит выходы шины адрес-данные и выходы линий управления в третье состояние. Это продолжается на протяжении действия сигнала *CLR*.

В минимальном режиме работы МП входы *STB* и *HLDA* во время действия сигнала *CLR* переводятся в пассивное состояние. В максимальном режиме сигналы *RQ/EO* и *RQ/EI* поддерживаются в пассивном состоянии, а сигналы *QSO* и *QSI* указывают на отсутствие операций.

Если сигнал *HLD* в минимальном режиме или сигналы *RQ/EO* и *RQ/EI* в максимальном активны сразу после окончания сигнала *CLR*, то они будут восприняты МП перед выборкой первой команды. Другими словами, после режима установки микросхема перейдет в состояние захвата или разрешения доступа к шинам.

Если на вход *NMI* поступит сигнал высокого уровня раньше, чем через девять тактов после окончания сигнала *CLR*, то МП будет выполнять очередную команду и только потом перейдет к обработке прерывания.

При выполнении команды HLT («Останов») МП переходит в состояние останова, которое характеризуется прекращением всех действий МП до появления внешних запросов прерывания или сигнала *CLR*. В этом режиме МП не выдает никаких сигналов, шина адрес-данные находится в третьем состоянии, которое можно использовать, например, при выявлении неисправности системы. Переход в состояние останова в минимальном режиме осуществляется по выдаче сигнала *STB* после выполнения команды HLT.

В максимальном режиме работы при выполнении команды HLT МП выдает комбинацию сигналов *ST0* = «0», *ST1* и *ST2* = = «1». Эта комбинация указывает на то, что МП перешел в состояние останова.

При работе в максимальном режиме МП выдает 8 сигналов состояния (*ST0* ... *ST7*), которые могут быть использованы внешними устройствами. Сигналы *ST0* ... *ST2* определяют тип цикла обмена по шине *AD* (табл. 5.6). Эти сигналы должны расшифровываться внешней схемой.

Сигналы состояния *ST3* и *ST4* определяют номер сегментного регистра, используемого в текущем цикле обмена (табл. 5.7).

Сигнал *ST5* индицирует состояние внутреннего триггера разрешения прерывания. Взведение этого триггера разрешает выполнение процедуры прерывания (по сигналу *INT*). Сигнал *ST6* всегда находится в состоянии низкого уровня. Сигнал *ST7* является вспомогательным, его состояние в процессе работы МП не определено.

В МП осуществляется предварительная выборка команд из внешней памяти и организация внутренней очереди длиной в 6 байт. Выборка кода команды в МП чаще всего происходит из очереди. Если все команды из очереди будут выполнены, МП

Таблица 5.6. **Типы циклов шины адрес-данные**

Сигналы состояния			Тип цикла шины
ST2	*ST1*	*ST0*	
0	0	0	Подтверждение прерывания
0	0	1	Чтение УВВ
0	1	0	Запись в УВВ
0	1	1	Останов
1	0	0	Выборка команды
1	0	1	Чтение памяти
1	1	0	Запись в память
1	1	1	Комбинация не используется

Таблица 5.7. **Назначение сигналов состояния *ST3*, *ST4***

Сигналы состояния		Сегментный регистр
ST4	*ST3*	
0	0	*ES*
0	1	*ES*
1	0	*CS* либо ввод-вывод при прерывании
1	1	*DS*

перейдет к организации новой очереди. Предварительная очередь дополняется новыми кодами команд, если в ней освобождаются два байта. В большинстве случаев очередь содержит не менее одного байта. Поэтому МП практически не тратит время на ожидание выборки команды.

В максимальном режиме МП выдает информацию о состоянии очереди команд (по выходам *QS0* и *QS1*). В табл. 5.8 представлены возможные комбинации сигналов *QS0* и *QS1*.

Сигналы *QS0* и *QS1* используются внешними устройствами (например, сопроцессорами) для организации приема от МП команды или операнда.

Рассмотрим подробнее процедуры прерывания. Вывод *INT* МП подключается к выходу контроллера прерываний, который, в свою очередь, подключен к устройствам, требующим прерывания. Запрос прерывания от контроллера прерываний может быть замаскирован. При этом внутренний триггер разрешения будет сброшен. Обработка маскируемого прерывания начинается по окончании последнего такта текущей команды. В связи с этим активный сигнал на входе *INT* должен держаться в течение такта, предшествующего концу команды. Если во время выполнения подпрограммы обработки текущего запроса прерывания поступит новый запрос, то последний будет игнорироваться МП до того момента, пока подпрограмма не восстановит триггер разрешения прерывания.

Следует отметить, что до окончания выполнения текущей команды действий по обслуживанию прерываний не производится.

Таблица 5.8. **Состояние очереди команд МП**

QS0	*QS1*	Состояние очереди
0	0	Нет выборки кодов из очереди
0	1	Первый байт команды выбран
1	0	Пуста
1	1	Выбран последний байт команды

Если на входе INT активный уровень сигнала и взведен триггер разрешения прерывания, то МП выполняет процедуру подтверждения прерывания. Так как сигнал INT не запоминается в МП, то его надо поддерживать в активном состоянии до начала процедуры (цикла) подтверждения прерывания. Этот цикл начинается в ответ на маскируемые прерывания. МП вырабатывает два последовательных цикла шины, разделенных холостыми тактами Т5. В каждом цикле МП выдает сигнал $INTA$. В этом случае системные устройства не должны выдавать информацию на шину без сигнала R от МП. Если в течение циклов подтверждения прерывания поступит запрос захвата (HLD или RQ/EI), то последний не будет воспринят до завершения этих циклов. В максимальном режиме для этого выдается сигнал $LOCK$, который указывает сопроцессорам системы, что захват шины запрещен.

Первый цикл подтверждения прерывания указывает контроллеру прерываний, что запрос принят. Во втором цикле контроллер выдает на шину данных байт типа прерываний (0 ... 255). МП по этому байту вызывает подпрограмму обслуживания прерывания данного типа. Таким образом, рассмотрены маскируемые прерывания, запросы от которых поступают по входу INT. Запросы, поступающие по входу NMI, имеют наивысший приоритет и относятся к немаскируемым прерываниям. Эти прерывания используют для обнаружения ошибок в памяти, ошибок четности шины, для обнаружения угрозы отключения питания.

Реакция на немаскируемые прерывания в МП начинается с момента прихода положительного фронта сигнала NMI. Это приводит к безусловному переходу на подпрограмму обслуживания. Сигнал на входе NMI должен поддерживаться в активном состоянии более двух тактов.

Немаскируемым прерываниям присвоен тип прерывания 2, поэтому нет необходимости передачи в МП кода типа прерывания, а значит, не требуется цикла подтверждения прерывания.

Внутренние прерывания можно разделить на три класса: предустановленные, аппаратные и программные. Первые возникают при появлении внутренних условий в МП.

Рассмотрим прерывания типов 0 ... 4. Прерывание типа 0 (ошибка деления) возникает в случае, когда результат превышает разрядность приемника. Примером этому может служить деление на нуль. Прерывание этого типа немаскируемо.

Прерывание типа 1 (пошаговый режим) возникает после окончания выполнения одной команды при условии взведения внутреннего триггера пошагового режима TF = «1». Последний может быть установлен путем записи в стек содержимого регистра признаков с последующей его выборкой. Прерывание этого типа немаскируемо.

Прерывание типа 2 обладает самым высоким приоритетом среди аппаратных прерываний и происходит по входу NMI.

Прерывание типа 3 называют однобайтным прерыванием и используют для «приостанова» в отладочных режимах. Это прерывание немаскируемо.

Прерывание типа 4 является прерыванием по переполнению (разряд *OF* установлен в «1») и реализуется при выполнении команды INTO. Прерывание немаскируемо.

Прерывания типов 0 и 2 возникают без вмешательства программиста, прерывания типов 1, 3, 4 требуют вмешательства программиста. Прерывание типа 0 обусловлено выполнением команд DIV или IDIV, типа 4 — команды INTO.

Рассмотрим программные прерывания. Пользователь может вызвать программным способом прерывание по команде INT. Второй байт этой команды указывает тип прерывания. Прерывания по команде INT не маскируются внутренним признаком разрешения прерывания. После окончания обслуживания прерывания необходимо выполнить команду возврата IRET, которая восстановит регистры МП для выполнения основной программы. Программные прерывания немаскируемы и не вызывают циклов подтверждения прерывания. Они инициируют передачу управления программе обслуживания в конце той команды, во время которой появляются. Адрес начала программы обработки прерывания находится в старшем слове вектора прерывания, а код типа прерывания — в младшем. Если в МП поступают одновременно несколько запросов, то он активизирует обслуживание в соответствии с их приоритетом. Так сигналы по входу *NMI* прервут маскируемое прерывание. Программные и внутренние запросы прервут начавшуюся процедуру прерывания. Необходимо следить за тем, чтобы внутри обслуживания прерывания не появились запросы прерывания того же типа.

Все процедуры прерывания должны заканчиваться командой IRET (возврат из прерывания). По этой команде три верхних слова из стека рассылаются в регистры *IP*, *CS* и РП. Таким образом управление будет передано команде, перед которой произошло прерывание.

Цикл подтверждения прерывания, как было указано, вырабатывается в ответ на прерывания, поступившие по входу *INT*. Этот цикл состоит из двух частей, разделенных двумя холостыми тактами. Во время действия цикла выдается сигнал *INTA*. МП не выдает адрес, но продолжает вырабатывать сигнал *STB*, который удобно использовать в сочетании с сигналами контроллера прерываний. Первый цикл свидетельствует о начале подтверждения прерывания и позволяет системе подготовиться к появлению в следующем цикле байта типа прерывания. Код типа прерывания должен быть передан в МП по младшей половине шины данных.

Отличие последовательности подтверждения прерывания МП К1810ВМ86 в том, что МП не нуждается в дополнительных командах, в то время как для микросхемы КР580ИК80 требуются команды RST или CALL.

В минимальном режиме в цикле подтверждения прерывания сигнал *M/IO* будет низкого уровня, свидетельствуя о процедуре ввода-вывода. С такта T2 первого цикла будет заблокирована возможность захвата шины. Сигнал готовности выполняет те же функции, что и для микросхемы КР580ИК80.

Вектор прерывания состоит из четырех байтов: два байта указателя команды, два байта сегментного регистра кода.

МП засылает в стек признаки и текущее содержимое указателя команд *IP* и сегментного регистра кода *CS*.

В указанные регистры засылаются новые значения, а триггеры пошагового режима и разрешения прерывания сбрасываются. Этим самым МП запрещается реагировать на последующие аппаратные прерывания в течение выполнения подпрограммы текущего обслуживания до тех пор, пока сама подпрограмма не установит триггер разрешения прерывания *IF* в состояние «1».

При формировании 20-разрядного адреса величина смещения вектора складывается с нулем, а сигналы состояния *ST4* и *ST3* («1» и «0» соответственно) указывают на отсутствие доступа к сегментным регистрам. Действия МП, связанные с подтверждением прерывания, сводятся к следующему:

выполнение двух циклов подтверждения прерывания;

считывание следующего указателя команд *IP* из зоны памяти;

считывание нового значения сегментного регистра *CS*;

запись в стек регистра признаков;

запись прежнего значения регистра *CS* в стек;

выбор кода операции первой команды подпрограммы обслуживания прерывания;

запись прежнего значения регистра *IP* в стек.

После последнего действия УСШ возобновляет предварительную выборку команд и формирует из них очередь. Уровень сигнала состояния *ST5* становится низким во втором такте, после считывания нового значения регистра *CS*.

Между окончанием выполнения текущей команды и началом выполнения подпрограммы обслуживания прерывания проходит 61 такт. При программных прерываниях эта величина составляет 51 такт для команд INT и пошагового режима, 52 такта для команды INT3, 53 такта для команды INT0.

Возможности МП позволяют создавать на его основе многопроцессорные системы. Так, в МП реализованы две версии мультиплексирования внешней (общей) шины: разделение шины и взаимное исключение активных устройств на ней. Разделение шины во времени между процессорами выполняется благодаря возможности предоставления шины.

Возможность взаимного исключения активных устройств реализуется сигналом *LOCK*, который может блокировать выходные каскады в возможно активных устройствах. Устройство сопряжения канала вырабатывает в максимальном режиме сигнал *LOCK* всякий раз, когда выполняется команда с префиксом *LOCK*

(префикс блокировки). Блок обработки декодирует префикс блокировки и сообщает о необходимости активизации сигнала *LOCK* в следующем такте. Сигнал поддерживается в активном состоянии в течение выполнения команды, следующей за командой с префиксом *LOCK*. При этом блокируется удовлетворение запросов прерывания, однако указанные запросы запоминаются. Если имеются две последовательные команды, каждой из которых предшествует *LOCK*, то между командами будет существовать незаблокированный промежуток.

В минимальном режиме МП не вырабатывает сигнала *LOCK*.

Сигнал *LOCK* может использоваться в многопроцессорных системах для координации доступа к совместным ресурсам (буферам, указателям). В противном случае возможна выборка одним процессором «чужих» данных.

Доступом к совместно используемым ресурсам можно управлять с помощью префикса *LOCK* и команды XCHG.

Под понятием «семафор» подразумевается программно управляемый признак, способный указывать доступность в данный момент времени совместно используемых ресурсов. Если ресурс доступен, то значение семафора «0».

Между процессорами устанавливается такая дисциплина общения, при которой во время доступа к ресурсу семафор взведен, по окончании доступа — сброшен. По команде XCHG процессор получит текущее значение семафора и установит его (при возможности доступа) в состояние «Занят». Однако при выполнении команды XCHG требуются два цикла шины. Между этими циклами другой процессор может захватить шину и получить доступ к семафору. Во избежание такой ситуации необходимо поставить команду с префиксом *LOCK* перед командой XCHG.

Цикл доступа к шине включает три фазы: запрос доступа к шине, разрешение доступа и освобождение шины. Сопроцессор сигналом *RQ*/*EI* запрашивает возможность доступа к шине. В конце текущего цикла шины или в холостом такте МП выдает (тем же сигналом) разрешение доступа и переводит свои выходные каскады шины *AD* в третье состояние. МП продолжает выполнение внутренних операций без выхода на внешнюю шину. В третьей фазе цикла доступа к шине сопроцессор, выполнивший процедуру доступа, вновь выдает сигнал *RQ*/*EI*, свидетельствующий о конце цикла захвата. Сигналы *RQ*/*EO* и *RQ*/*EI* отличаются уровнями приоритета (первый из них имеет высший приоритет). В минимальном режиме сигналы *RQ*/*EO*, *RQ*/*EI* и *HLD* имеют более высокий приоритет, чем ожидаемое прерывание.

Время реакции МП от момента запроса до разрешения для обычной команды может составлять от трех (без обращения к памяти) до десяти тактов. Сигнал *LOCK* вырабатывается либо в цикле подтверждения прерывания, либо при выполнении команды с префиксом *LOCK*. При выполнении команды проверки семафора XCHG максимальная задержка составляет 39 тактов.

После снятия сигнала *CLR* МП сразу может воспринимать запросы доступа к шине. Это облегчает передачу управления в начальный момент времени в ячейки памяти по адресу $FFFF0H_{16}$. В минимальном режиме к аналогичному результату приводит подача запросов *HLD* или *HLDA* в первом или втором такте после снятия сигнала *CLR*.

На рис. 5.2 приведена структура шин МП. В МП имеется два типа шин — *локальная* и *общая системная*. Процессоры всегда подключаются к локальной шине, а *ЗУ* и *УВВ* — к системной.

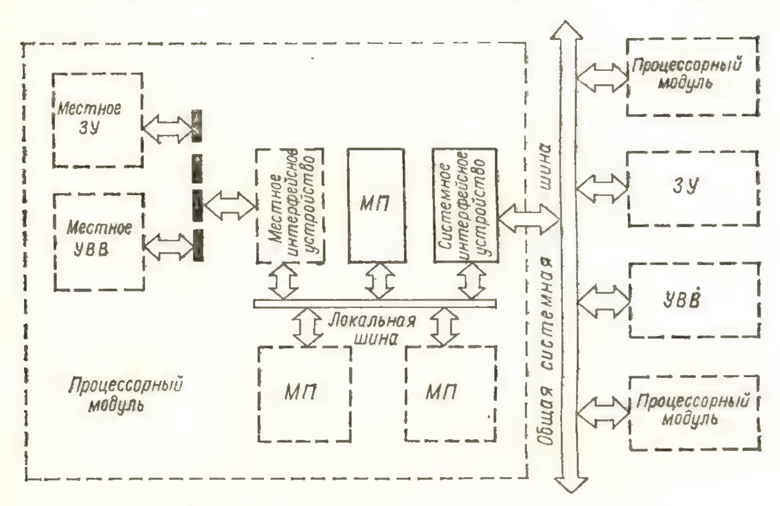


Рис. 5.2. Общая структура шин микросхемы К1810ВМ86

Локальную шину можно разделить во времени как между независимыми процессорами, так и сопроцессорами. Процессоры в локальной шине используют один и тот же интерфейс, а шина отличается регулярной конфигурацией и простотой организации.

Системная шина связана с локальной через *системное интерфейсное устройство*. Системная шина включает пять групп линий: адреса, данных, управления, прерывания, арбитража. Системная шина обычно связана с блоками *ЗУ* и устройствами ввода-вывода *УВВ*.

Линии адреса и данных демультиплексированы. Линии управления включают стандартные сигналы чтения, записи и т. д.

Системное интерфейсное устройство зависит от сложности проектируемой системы: например, от таких параметров, как количество регистров, разрядность линий данных, необходимость разделения шины во времени. Шина подчинена асинхронной дисциплине обмена. В течение такого цикла обмена выдается, например, адрес ячейки *ЗУ* или *УВВ*, затем сигнал характера обращения (чтение или запись), а затем данные поступают в вы-

бранное устройство. В конце цикла принимающее устройство фиксирует данные, а выдающее устройство снимает их.

Каждый цикл шины состоит, как минимум, из четырех машинных тактов *T1* ... *T4*. В такте *T1* МП выдает на шину адрес внешнего устройства. В такте *T2* шина *AD* находится в третьем состоянии в том случае, если выполняется цикл «Чтение». Если выполняется цикл «Запись» данных во внешнее устройство, то выдача данных из МП происходит в тактах *T2* ... *T4* (рис. 5.3). Данные для ввода в МП в цикле «Чтение» появляются в *T3*, а в *T4* они

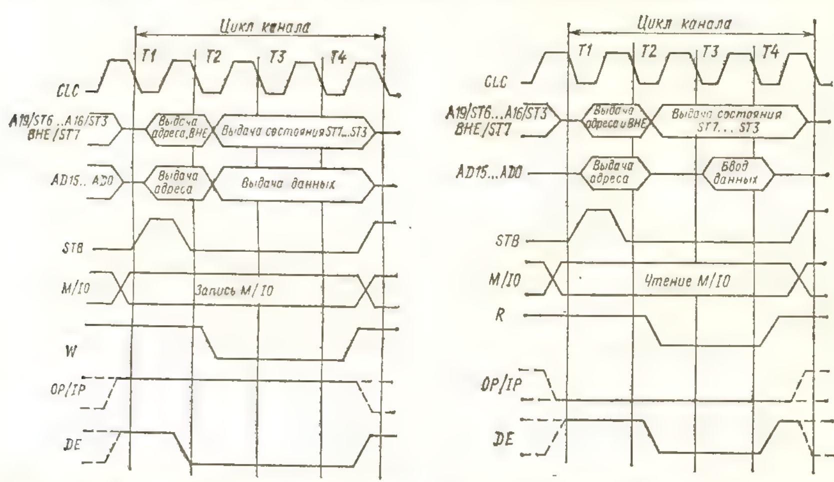


Рис. 5.3. Временна́я диаграмма цикла «Запись» микросхемы К1810ВМ86

Рис. 5.4. Временна́я диаграмма цикла «Чтение» микросхемы К1810ВМ86

сбрасываются (рис. 5.4). В циклах «Чтение» и «Запись» данные на шине стробируются сигналом *DE*. Сигналы *R* («Чтение») и *W* («Запись») появляются в соответствующих циклах в тактах *T2* ... *T4*.

Во время выполнения циклов обмена могут возникнуть такты ожидания, обычно между *T3* и *T4*. Их появление обусловлено наличием медленно действующих устройств в системе. Такты ожидания (T_W) инициируются сигналом *RDY* («Готовность»).

В процессе выполнения команд между циклами шины могут появиться холостые такты (*T*5), в течение которых УСШ не активно. Например, для команды умножения MUL количество холостых тактов шины может составлять 52 ... 69 тактов, пока устройство обработки МП не завершит выполнение текущей команды.

МП предоставляет возможность следить за ходом выполнения цикла шины, а также идентифицировать тип цикла шины (сигналы *ST0* ... *ST2*). Кроме того, с помощью сигналов *QS0* и *QS1* внешние устройства могут следить за состоянием внутренней очереди команд МП.

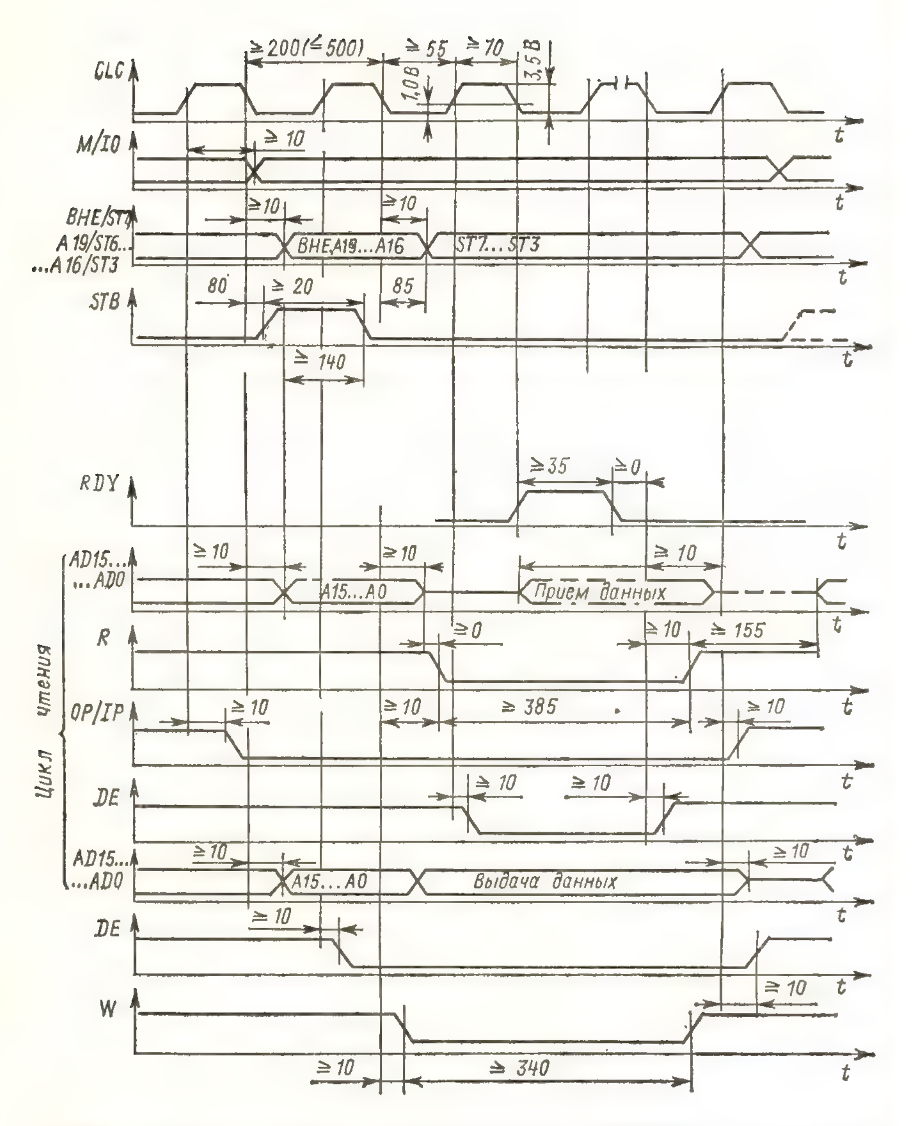


Рис. 5.5. Временна́я диаграмма работы микросхемы К1810ВМ86 при выполнении циклов «Чтение» и «Запись» в минимальном режиме

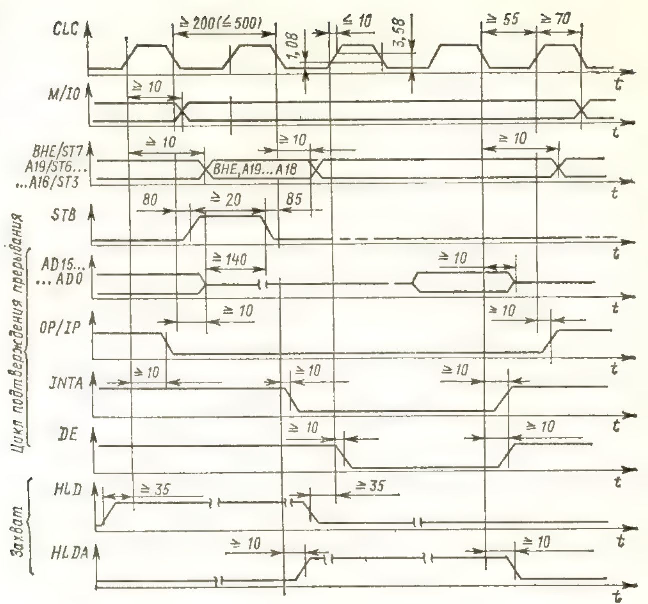


Рис. 5.6. Временная диаграмма работы микросхемы К1810ВМ86 при «Подтверждении прерывания» и «Захвате» в минимальном режиме

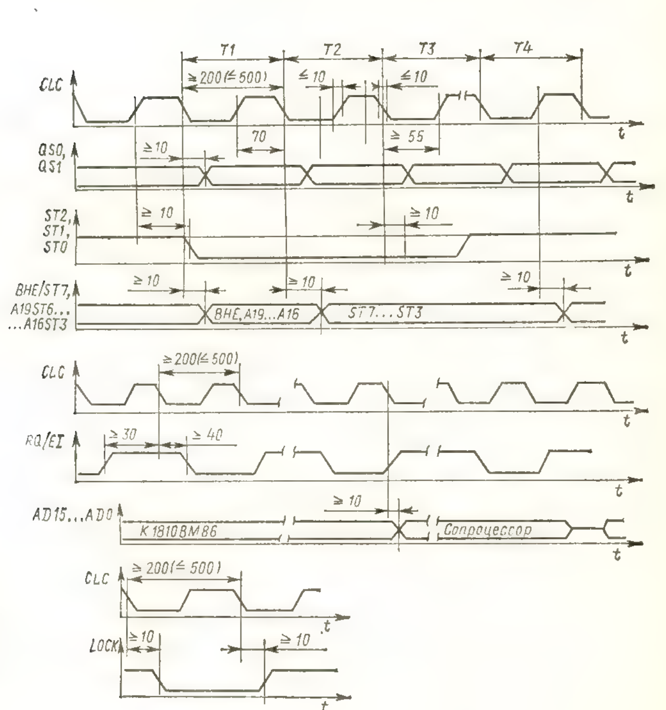


Рис. 5.7. Временная диаграмма работы микросхемы К1810ВМ86 в максимальном режиме

Таблица 5.9. Статические параметры микросхемы К1810ВМ86

Параметр, единица измерения	Обозначение	Значение		Режим измерения
		мин.	макс.	
Напряжение источника питания, В	U_{CC}	4,75	5,25	—
Входное напряжение «0», В	U_{IL}	—	0,4	—
Входное напряжение «1», В	U_{IH}	2,4	—	—
Выходное напряжение «0», В	U_{OL}	—	0,45	$I_{OL} = 20$ мА
Выходное напряжение «1», В	U_{OH}	2,4	—	$I_{OH} = 0,4$ мА
Входное напряжение «1» на входе *CLC*, В	U_{IH}	4,0	—	—
Ток потребления, мА	I_{CC}	—	340	T = +25 °C
Выходной ток в состоянии «Выключено», мкА	I_{OZ}	—	±10	—
Входная емкость, пФ	C_I	—	15	—
Емкость выводов типа вход-выход, пФ	$C_{I\text{-}O}$	—	20	—
Емкость нагрузки, пФ	C_L	20	200	$f = 1$ МГц

Работу шины *AD* МП можно разделить на шесть циклов: «Выдача адреса», «Чтение», «Запись», «Подтверждение прерывания», «Ожидание готовности», «Захват».

Выдача адреса на шину *AD* сопровождается сигналом *STB*. Внешние устройства могут принимать (защелкивать) адрес по фронту сигнала *STB*.

Цикл «Чтение» состоит из подготовки шины, выдачи сигнала («Чтение»), а также выдачи сигналов разрешения и направления передачи данных *DE* и *OP/IP*. Сигнал направления передачи в цикле «Чтение» устанавливается в «0» в такте *T1* и сбрасывается в такте *T4*. Этот же сигнал (*OP/IP*) в цикле «Запись» устанавливается в состояние «1» в такте *T1* и сбрасывается в «0» в такте *T4*.

Сигнал *DE* устанавливается в состояние «0», стробируя данные на шине *AD*. В цикле «Запись» сигнал *DE* устанавливается в состояние «0» в такте *T1*, а сбрасывается в такте *T4*. В цикле «Чтение» сигнал *DE* взводится в состояние «0» в такте *T2*, а сбрасывается в такте *T4*. В максимальном режиме работы сигнал *M/IO* идентифицирует обращение к ЗУ или устройствам ввода-вывода.

На временны́х диаграммах рис. 5.5 ... 5.7 иллюстрируется функционирование МП в различных режимах (циклах шины). Статические параметры микросхемы К1810ВМ86 приведены в табл. 5.9.

5.3. СИСТЕМА КОМАНД МИКРОСХЕМЫ К1810ВМ86

Система команд МП К1810ВМ86 позволяет повысить производительность микропроцессорных систем, сократить трудоемкость их программирования, обеспечить проектирование многопроцес-

сорных систем и преемственность с широко используемыми системами на основе БИС серии КР580. Система команд этого МП содержит традиционные группы команд-пересылок (арифметические, поразрядной обработки данных, перехода и управления процессором) и группу команд обработки строк, обеспечивающих программирование и повышающих производительность обработки текстовых данных. Программно-доступные элементы адресуются с помощью неявной, непосредственной, прямой регистровой, косвенной регистровой и косвенной адресациями.

Как было отмечено, в состав программно-доступных регистров МП К1810ВМ86 входят 16-разрядные РОН (*AX*, *BX*, *CX*, *DX*, *SP*, *BP*, *SI*, *DI*), 16-разрядные сегментные регистры (*CS*, *DS*, *SS* и *ES*), 16-разрядный регистр адреса команд *IP* и регистр признаков *F*. РОН *AX*, *BX*, *CX* и *DX* могут использоваться как восемь 8-разрядных регистров *AL*, *BL*, *CL*, *DL*, *AH*, *BH*, *CH* и *DH*.

Регистр *F* содержит признаки результат (*CF* — признак переноса, *PF* — четности, *CnF* — вспомогательного переноса, *ZF* — нуля, *SF* — знака и *OF* — переполнения) и признаки управления (*IF* — признак разрешения прерывания, *TF* — шагового режима и *DF* — направления).

Общий объем адресуемой памяти МП К1810ВМ86 — 1 Мбайт. Структурные единицы памяти — байт, слово (2 байта), двойное слово и сегмент (64 байта). Адресом байта, слова и двойного слова может быть любое 20-разрядное двоичное число. Адрес сегмента памяти всегда содержит в четырех младших разрядах нули. 20-разрядный адрес структурных единиц памяти не является программно-доступным; старшие 16 разрядов адреса сегмента, называемые базой сегмента и располагаемые в сегментных регистрах — программно-доступны. Для адресации остальных структурных единиц памяти в программах используется логический адрес (указатель), содержащий две 16-разрядные компоненты: базу сегмента, помещаемую в сегментный регистр, и относительный адрес (смещение) в сегменте.

В один и тот же момент программе доступны четыре текущих сегмента по 64 Кбайт каждый: сегмент кодов, адресуемый регистром *CS*, данных, адресуемый регистром *DS*, стека, адресуемый регистром *SS*, и вспомогательный сегмент, адресуемый регистром *ES*. Сегменты могут быть разнесены, перекрываться и совпадать. Объем и типы сегментов памяти обеспечивают преемственность с МП КР580ИК80, в котором выделение сегментов кодов, данных и стека осуществляется на программном уровне.

Команды МП К1810ВМ86 занимают в памяти от одного до шести байт. Первый байт команды содержит код операции. В некоторых командах код операции размещается и во втором байте.

Одноразрядное поле *W* имеется в подавляющем большинстве команд и определяет разрядность операндов. Если $W = 0$, то операнды 8-разрядные; если $W = 1$, то 16-разрядные. Поля *mod* и *r/m* второго байта команды определяют один из 24 режимов

Таблица 5.10. **Основные команды микросхемы К1810ВМ86**

Номер команды	Мнемоника	Действие команды
1	*MOV*	$E \leftarrow R,\ R \leftarrow E,\ E \leftarrow D$
2	*PUSH*; *POP*	$St \leftarrow E16;\ E16 \leftarrow St$
3	*XCHG*	$E \leftrightarrow R$
4	*MOV*	$R \leftarrow D,\ A \leftarrow M\ (B3\ \&\ B2),\ M\ (B3\ \&\ B2) \leftarrow$ $\leftarrow A$
5	*PUSH*; *POP*	$St \leftarrow R16;\ R16 \leftarrow St$
6	*XCHG*	$AX \leftrightarrow R16$
7	*MOV*	$R_{seg} \leftarrow E\ (R_{seg} \neq CS),\ E \leftarrow R_{seg}$
8	*PUSH*; *POP*	$St \leftarrow R_{seg};\ R_{seg} \leftarrow St\ (R_{seg} \neq CS)$
9	*LEA*	$R16 \leftarrow EA$
10	*LDS*	$DS \leftarrow M16\ (EA + 2),\ R16 \leftarrow M16\ (EA)$
11	*LES*	$ES \leftarrow M16\ (EA + 2),\ R16 \leftarrow M16\ (EA)$
12	*PUSHF*; *POPF*	$St \leftarrow F,\ F \leftarrow St$
13	*LAHF*, *SAHF*	$AH \leftarrow F;\ F \leftarrow AH$
14	*IN*	$A \leftarrow Port\ (B2),\ A \leftarrow Port\ (DX)$
15	*OUT*	$Port\ (B2) \leftarrow A,\ Port\ (DX) \leftarrow A$
16	*XLAT*	$AL \leftarrow M\ (BX + AL)$
17	*ADD*	$E \leftarrow E + R,\ R \leftarrow R + E,\ E \leftarrow E + D$ $E \leftarrow E + Dext$
18	*ADC*	$E \leftarrow E + CF + R,\ R \leftarrow R + CF + E,$ $E \leftarrow E + CF + D,\ E \leftarrow E + CF + D$
19	*INC*	$E \leftarrow E + 1$
20	*AAA*, *DAA*	Коррекция результата сложения распакованных и упакованных двоично-десятичных чисел
21	*SUB*	$E \leftarrow E - R,\ R \leftarrow R - E,\ E \leftarrow E - D,$ $E \leftarrow E - Dext$
22	*SBB*	$E \leftarrow E - CF - R,\ R \leftarrow R - CF - E,$ $E \leftarrow E - CF - D,\ E \leftarrow E - CF - Dext$

Номер команды	Мнемоника	Действие команды
23	*DEC*	$E \leftarrow E - 1$
24	*NEG*	$E \leftarrow 0 - E$
25	*CMP*	$F \leftarrow f\,(E - R)$, $F \leftarrow f\,(R - E)$, $F \leftarrow f\,(E - D)$, $F \leftarrow f\,(E - Dext)$
26	*AAS, DAS*	Коррекция результата вычитания
27	*ADD, ADC*	$A \leftarrow A + D$, $A \leftarrow A + CF + D$
28	*INC, DEC*	$R \leftarrow R + 1$, $R \leftarrow R - 1$
29	*CUB, SBB*	$A \leftarrow A - D$, $A \leftarrow A - CF - D$
30	*CMP*	$F \leftarrow f\,(A - D)$
31	*MUL, IMUL*	$\underbrace{AX \leftarrow AL \times E;}_{W=0}\quad \underbrace{DX\ \&\ AX \leftarrow AX \times E}_{W=1}$
32	*DIV, IDIV*	$\left.\begin{array}{l} AL \leftarrow AX : E \\ AH \leftarrow mod\,(EX,\ E) \end{array}\right\}\ W = 0$ $\left.\begin{array}{l} AX \leftarrow (DX\ \&\ AX) : E \\ DX \leftarrow mod\,(DX\ \&\ AX),\ E \end{array}\right\}\ W = 1$
33	*AAM*	Преобразование двоичного числа в регистре *AL* в распакованное двоично-десятичное число в регистрах *AL* и *AH*
34	*AAD*	Преобразование распакованного двоично-десятичного числа в *AH* и *AL* в двоичное число в *AL*
35	*CBW*	$AH \leftarrow AL\ ext$
36	*CWD*	$DX\ \&\ AX \leftarrow AX\ ext$
37	*NOT*	$E \leftarrow \overline{E}$
38	*AND*	$E \leftarrow E \wedge R$, $R \leftarrow R \wedge E$, $E \leftarrow E \wedge D$
39	*OR*	$E \leftarrow E \vee R$, $R \leftarrow R \vee E$, $E \leftarrow E \vee D$
40	*XOR*	$E \leftarrow E \oplus R$, $R \leftarrow R \oplus E$, $E \leftarrow E \oplus D$
41	*TEST*	$F \leftarrow\ (E \wedge R)$, $F \leftarrow f\,(R \wedge E)$, $F \leftarrow f\,(E \wedge D)$

Номер команды	Мнемоника	Действие команды
42	*AND*; *OR*	$A \leftarrow A \wedge D$, $A \leftarrow A \vee D$
43	*XOP*; *TEST*	$A \leftarrow A \oplus D$, $F \leftarrow f\,(A \wedge D)$
44	*SHL*	$E \leftarrow E \times 2$, $E \leftarrow E \times 2^{cx}$
45	*SAR*	$E \leftarrow E : 2$, $E — E : 2^{cx}$, — деление целых чисел
46	*SHP*	$E \leftarrow E : 2$, $E \leftarrow E : 2^{cx}$, — деление порядковых чисел
47	*ROL*	Сдвиг циклический влево 1 или *CL* раз содержимого *E*
48	*ROR*	Сдвиг циклический вправо 1 или *CL* раз содержимого *E*
49	*RCL*	Сдвиг циклический влево 1 или *CL* раз содержимого *E* совместно с *CF*
50	*RCR*	Сдвиг циклический вправо 1 или *CL* раз содержимого *E* совместно с *CF*
51	*JMP*	Межсегментная прямая адресация $IP \leftarrow B3\ \&\ B2$; $C \leftarrow B5\ \&\ B4$ Межсегментная косвенная $CS\ \&\ IP \leftarrow M16\,(EA + 2)\ \&\ M16\,(EA)$ Внутрисегментная прямая $IP \leftarrow IP + B3\ \&\ B2$ Внутрисегментная прямая короткая $IP \leftarrow IP + B2\,ext$ Внутрисегментная косвенная $IP \leftarrow E$
52	*CALL*	Межсегментная прямая адресация $St \leftarrow CS$, $St \leftarrow IP$, $IP \leftarrow B3\ \&\ B2$, $CS \leftarrow B5\ \&\ B4$ Межсегментная косвенная $St \leftarrow CS$, $St \leftarrow IP$ $CS\ \&\ IP \leftarrow M16\,(EA + 2)\ \&\ M16\,(EA)$ Внутрисегментная прямая $St \leftarrow IP$, $IP \leftarrow IP + B3\ \&\ B2$ Внутрисегментная косвенная $St \leftarrow IP$, $IP \leftarrow E$
53	*RET*	Межсегментная адресация $IP \leftarrow St$, $CS \leftarrow St$ Внутрисегментная $IP \leftarrow St$

Номер команды	Мнемоника	Действие команды
54	*INT*	$St \leftarrow F$, $St \leftarrow CS$, $St \leftarrow IP$ CS & $IP \leftarrow Port\,(B2)$, $IF \leftarrow 0$, $TF \leftarrow 0$
55	*INT3*	Отличие от *INT*: CS & $IP \leftarrow Port\,(B3)$
56	*INT0*	Отличие от *INT*: если $OF = 1$, то CS & $IP \leftarrow Port\,(B4)$
57	*IRET*	$IP \leftarrow St$, $CS \leftarrow St$, $F \leftarrow St$
58	*STC, CLC, CMC*	$CF \leftarrow 1$, $CF \leftarrow 0$, $CF \leftarrow \overline{CF}$
59	*STD, CLD*	$DF \leftarrow 1$, $DF \leftarrow 0$
60	*STI, CLI*	$IF \leftarrow 1$, $IF \leftarrow 0$

Таблица 5.11. **Команды условных переходов микросхемы К1810ВМ86**

Мнемокод	Отношение	Тип чисел	Условие
JE	$a = b$	Целые, порядковые	$ZF = 1$
JZ	$a = 0$		
JNE	$a \neq b$		$ZF = 0$
JNZ	$a \neq 0$		
JB	$a < b$	Порядковые	$CF = 1$
JNAE	$a \geqslant b$		
JC	—		
JNB	$a > b$		$CF = 0$
JAE	$a \leqslant b$		
JNG	—		
JG	$a > b$	Целые	$((SF \oplus OF) \vee ZF) = 0$
JNLE	$a \leqslant b$		

Продолжение табл. 5.11

Мнемокод	Отношение	Тип чисел	Условие
JGE	$a \geqslant b$	Целые	$(SF \oplus OF) = 0$
JNL	$a \not< b$		
JL	$a < b$		$(SF \oplus OF) = 1$
JNGE	$a \geqslant b$		
JLE	$a \leqslant b$		$((SF \oplus OF) \vee ZF) = 1$
JNG	$a > b$		
JA	$a > b$	Порядковые	$(CF \vee ZF) = 0$
JNBE	$a \leqslant b$		
JBE	$a \leqslant b$		$(CF \vee ZF) = 1$
JNA	$1 > b$		
JO	—	—	$OF = 1$
JNO	—	—	$OF = 0$
JS	—	—	$SF = 1$
JNS	—	—	$SF = 0$
JP, JPE	—	—	$PF = 1$
JNP, JPO	—	—	$PF = 0$
JCXZ	—	—	$CX = 0$

формирования относительного адреса операнда в сегменте данных или в сегменте стека. Поля *disp8* и *disp16* определяют одно- или двухбайтное смещение в команде. Если смещение однобайтное, то производится его знаковое расширение до двух байт (т. е. значение всех разрядов второго байта устанавливается равным значению старшего разряда исходного байта).

В табл. 5.10 и 5.11 приведена система команд МП К1810ВМ86. Приняты следующие обозначения: E — операнд в памяти или в РОН; R — операнд в РОН; R_{seg} — операнд в сегментном регистре; D — непосредственные данные в команде; A — если $W = 0$, то операнд в регистре AL, если $W = 1$, то операнд в регистре AX; St — стек; M (...) — операнд в памяти (в скобках

указывается смещение в сегменте); *Port* (...) — операнд в регистре внешнего устройства (в скобках адрес устройства); *EA* — исполнительный адрес операнда в памяти, определяемый полями *mod* и *r/m*; *ext* — знаковое расширение байта в слово или слова в двойное слово; & — объединение байт в слово, слово — в двойное слово; *B2* и *B3* — второй и третий байт команды; *f* (...) — функция определения значений признаков результата по результату операции, указанной в скобках.

Глава 6

МИКРОПРОЦЕССОРНЫЙ КОМПЛЕКТ СЕРИИ К588 (КР588)

6.1. ОБЩИЕ СВЕДЕНИЯ

Микропроцессорный комплект серии К588 создан на базе КМОП-технологии и предназначен для построения аппаратуры средней производительности. Отличительными особенностями микросхем комплекта являются малое потребление мощности от источника питания, асинхронный принцип обмена информацией, а также широкий набор периферийных БИС комплекта. В состав БИС комплекта входят, в частности, БИС управляющей памяти, кодировки которых полностью реализуют систему команд микроЭВМ «Электроника-60» и операции расширенной арифметики. Однако комплект серии К588 может реализовывать устройства не только с этой системой команд. Пользователь строит собственную микропрограммную память, реализующую практически любую систему команд [9, 34, 45, 46].

Микросхемы комплекта обеспечивают выходные сигналы, соответствующие уровням сигналов ТТЛ-микросхем. Однако ввиду того, что входные напряжения сигналов «1» микросхем комплекта превышают нижнюю границу ТТЛ-сигналов, стыковку по входам можно обеспечить ТТЛ-микросхемами с открытым коллектором и нагрузочными резисторами.

В состав комплекта входят БИС системного контроллера для сопряжения устройства на базе микросхем комплекта с каналом типа *Q*-шина, БИС аппаратного умножителя, магистрального приемопередатчика, буферного регистра, контроллера ЗУ и селектора адреса. Состав комплекта приведен в табл. 6.1.

Микросхемы комплекта обладают высокой помехоустойчивостью.

Комплект серии К588 (КР588) служит для построения систем управления периферийными устройствами. В основном его исполь-

Таблица 6.1. Состав серии К588 (КР588)

Тип микросхемы	Назначение	Краткая характеристика	Тип корпуса
К588ВУ2 (КР588ВУ2)	Управляющая память	Время цикла $\leqslant$0,8 мкс; $P_{CCS} \leqslant$ 0,5 мВт; $P_{CC} \leqslant$ 20 мВт; 12-разрядная микрокоманда; число БИС — 7	429.42-3 (2206.42-2)
К588ВС2 (КР588ВС2)	Арифметическое устройство	Время цикла (среднее) $\leqslant$1,82 мкс; $P_{CCS} \leqslant$ 1 мВт; $P_{CC} \leqslant$ 30 мВт; 16-разрядное	
К588ВГ1 (КР588ВГ1)	Системный контроллер	Время выдачи адреса $\leqslant$150 нс; $P_{CCS} \leqslant$ 0,5 мВт; $P_{CC} \leqslant$ 15,0 мВт	
К588ВР2	Умножитель	Время умножения $\leqslant$2,0 мкс; $P_{CCS} \leqslant$ 5,0 мВт; $P_{CC} \leqslant$ 35,0 мВт; 16-разрядный	4118.24-2
К588ВА1 (КР588ВА1)	Магистральный приемопередатчик	Время передачи $\leqslant$150 нс; $P_{CCS} \leqslant$ 0,50 мВт; $P_{CC} \leqslant$ 25,0 мВт; 8-разрядный	4119.28-4 (2121.28-4)
К588ИР1 (КР588ИР1)	Многофункциональный буферный регистр	Время записи $\leqslant$100 нс; Время считывания $\leqslant$ 150 нс; $P_{CCS} \leqslant$ 0,3 мВт; $P_{CC} \leqslant$ 10,0 мВт; 8-разрядный	
К588ВГ2 (КР588ВГ2)	Контроллер запоминающего устройства	Время выборки $\leqslant$ 200 нс; $P_{CCS} \leqslant$ 0,3 мВт; $P_{CC} \leqslant$ 15,0 мВт	4116.18-2 (2107.18-1)
К588ВТ1	Селектор адреса	Время выборки $\leqslant$ 300 нс; $P_{CCS} \leqslant$ 2,5 мВт; $P_{CC} \leqslant$ 30 мВт	429.42-3

зуют в устройствах, действующих в условиях сильных электромагнитных полей, в аппаратуре, работающей в режиме длительного автономного плавания и др.

Микросхемы выпускают в корпусах с горизонтальным (К588) и вертикальным (КР588) расположением выводов. Электрические параметры, а также функциональные возможности микросхем с индексами К и КР идентичны.

6.2. МИКРОСХЕМА К588ВУ2 (КР588ВУ2)

Представляет управляющую память микрокоманд (УП). Основой УП являются программируемые логические матрицы ПЛМ. Программирование УП выполняют на заводе-изготовителе масочным способом. В настоящее время выпускают семь типов УП — К588ВУ2-0001 ... К588ВУ2-0007. Кодировки микросхем УП полностью реализуют систему команд микроЭВМ «Электроника-60»

Рис. 6.1. Условное графическое обозначение микросхемы К588ВУ2

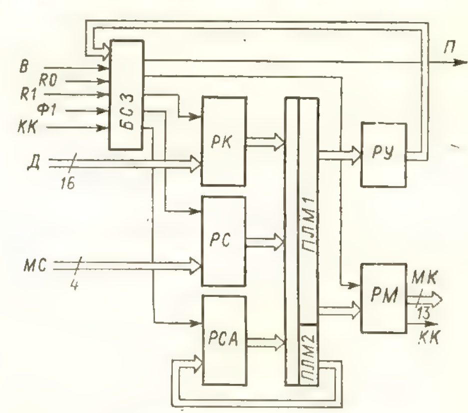


Рис. 6.2. Структурная схема микросхемы К588ВУ2

и операции расширенной арифметики. К последним относятся команды умножения, деления, сдвига. Кодировка К588ВУ2-0005 содержит микропрограммы работы микросхемы системного контроллера.

На рис. 6.1 и 6.2 изображены соответственно условное графическое обозначение и структурная схема БИС УП. В табл. 6.2 дано назначение выводов этой микросхемы.

В состав структурной схемы БИС УП входят следующие функциональные узлы:

регистр команд *РК*; принимает команду по шине команд *Д*;

программируемые логические матрицы *ПЛМ1* и *ПЛМ2*; в них запрограммированы не только информация о выбираемой

Таблица 6.2. **Назначение выводов микросхемы К588ВУ2 (КР588ВУ2)**

Номер вывода	Назначение	Разряд	Обозначение	Тип
1		0	МС0	
2	Магистраль состояния	1	МС1	
3		2	МС2	
4		3	МС3	
5		0	Д0	
6		1	Д1	
7		2	Д2	
8		3	Д3	
9		4	Д4	
10		5	Д5	
11		6	Д6	Вход
12		7	Д7	
13	Шина команд	8	Д8	
14		9	Д9	
15		10	Д10	
16		11	Д11	
17		12	Д12	
18		13	Д13	
19		14	Д14	
20		15	Д15	
21	Общий	—	*GND*	—
22		0	МК0	
23		1	МК1	
24		2	МК2	
25		3	МК3	
26		4	МК4	
27		5	МК5	
28	Магистраль микрокоманд	6	МК6	Вход
29		7	МК7	
30		8	МК8	
31		9	МК9	
32		10	МК10	
33		11	МК11	
34		12	МК12	
35	Не используется	—	—	—
36	Завершение выполнения микрокоманды	—	Ф1	Вход
37	Установка	1	*R*1	
38	Конец команды	—	КК	Вход-выход
39	Установка	0	*R*0	Вход
40	Команда принята	—	П	Выход
41	Команда выдана	—	В	Вход
42	Напряжение питания	—	U_{CC}	—

микрокоманде и адрес следующей микрокоманды, но и разряды, управляющие внутренними узлами БИС УП;

регистры состояния *РС* и следующего адреса *РСА*; принимают соответственно признаки результата выполнения операции и адрес следующей микрокоманды. Информация состояния поступает, как правило, из арифметического устройства и включает 4-разрядное слово-состояние, состоящее из признаков нулевого результата, переполнения, переноса из старшего знакового разряда (*Z*, *V*, *C*, *N*);

регистры управления *РУ* и микрокоманд *РМ*;

блок синхронизации *БСЗ*; вырабатывает управляющие сигналы записи во внутренние регистры БИС УП и сигнал квитирования *П*, свидетельствующий об окончании приема команды. На *БСЗ* поступают разряды регистра *РУ*.

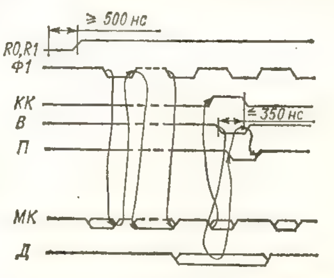


Рис. 6.3. Временна́я диаграмма работы микросхемы К588ВУ2

Функционирование БИС УП начинается с подачи на входы *R0* и *R1* отрицательных сигналов, которые переводят БИС в исходное состояние. Длительность сигналов *R0* и *R1* не менее 500 нс.

Работа микросхемы УП начинается по положительному фронту сигнала *R0* (либо *R1*). По отрицательному сигналу *В* происходит прием команды во входной регистр *РК*, а также информации состояния в регистр *РС*. После приема этой информации БИС УП выдает отрицательный сигнал *П*, свидетельствующий о том, что команда принята. Отрицательный фронт сигнала *П* инициирует положительный фронт сигнала *В*, который, в свою очередь, сбрасывает сигнал *П* в состояние «1».

Затем начинается выдача первой микрокоманды принятой команды. Положительный фронт сигнала *Ф1* инициирует выдачу очередной микрокоманды на магистраль *МК*. Микрокоманда может находиться на магистрали до отрицательного фронта сигнала *Ф1*. После этого происходит «подброс в единицу» магистрали *МК*. Если к этому времени сформирована следующая микрокоманда, то происходит ее запись в *РМ* с последующей выдачей. Одновременно с выдачей последней микрокоманды текущей команды выдается сигнал *КК*, сигнализирующий контроллеру (либо внешнему ЗУ) о необходимости подачи новой команды.

На рис. 6.3 приведена временна́я диаграмма работы БИС УП.

Реализация всей системы команд микроЭВМ «Электроника-60» и ограниченные возможности внутренних ПЛМ БИС УП привели к необходимости использования не менее пяти «зашивок» микрокоманд в БИС УП (0001 ... 0005). Первые четыре реализуют собственно команды «Электроники-60», пятая — управляет системным контроллером и разрешением прерываний. Четыре ми-

Таблица 6.3. Основные электрические параметры БИС УП

Параметр, единица измерения	Обозначение	Значение
Напряжение питания, В	U_{CC}	5 ± 5 %
Статическая потребляемая мощность, мВт	P_{CCS}	⩽0,5
Динамическая потребляемая мощность, мВт	P_{CC}	⩽20
Входное напряжение «0», В	U_{IL}	⩽0,4
Входное напряжение «1», В	U_{IH}	⩾4,1
Входной ток «1», мкА	I_{IH}	⩽10
Выходной ток «0», мА	I_{OL}	⩾0,8
Выходной ток «1», мА	I_{OH}	⩾0,04
Время цикла, мкс	t_{CY}	⩽0,8

кросхемы БИС УП (0001 ... 0004) соединяются параллельно (по выходам) и выдают информацию в арифметическое устройство (АУ); 12-разрядный код является собственно микрокомандой для АУ. Разряд 12 используется в качестве строба микрокоманды. Микросхема К588ВУ2-0005 разрядами 0 ... 6 вырабатывает сигналы разрешения прерывания, разряды 7 ... 12 управляют микросхемой К588ВГ1 (КР588ВГ1).

В табл. 6.3 приведены основные электрические параметры БИС УП.

6.3. МИКРОСХЕМА К588ВС2 (КР588ВС2)

Представляет БИС 16-разрядного арифметического устройства (БИС АУ). БИС АУ предназначена для построения процессоров микроЭВМ, а также центральных устройств обработки программируемых контроллеров.

БИС АУ осуществляет обмен информацией с внешними устройствами по 16-разрядной двунаправленной магистрали данных. Операции обмена информацией сопровождаются сигналами квитирования. Микросхема реализует асинхронную дисциплину обмена.

Назначение выводов БИС АУ представлено в табл. 6.4. Условное графическое обозначение и структурная схема БИС АУ изображены соответственно на рис. 6.4 и 6.5. В состав структурной схемы БИС АУ входят следующие функциональные узлы:

регистр микрокоманд *РМ* и дешифратор *ДШ*; предназначены для приема и дешифрации кода микрокоманды;

буферный регистр данных *БРг*; через *БРг* осуществляют связь между внутренней шиной данных *ВШД* и внешней магистралью данных *Д*;

арифметико-логическое устройство *АЛУ* (16-разрядное); в нем происходят основные действия, связанные с арифметико-логической обработкой данных. Прием информации в *АЛУ*, а также ее выдача осуществляются через *ВШД*;

Таблица 6.4. Назначение выводов микросхемы К588ВС2

Номер вывода	Назначение	Разряд	Обозначение	Тип
1	Шина микрокоманд	7	МК7	Вход
2		8	МК8	
3		9	МК9	
4		10	МК10	
5	Магистраль данных	0	Д0	Вход-выход
6		1	Д1	
7		2	Д2	
8		3	Д3	
9		4	Д4	
10		5	Д5	
11		6	Д6	
12		7	Д7	
13		8	Д8	
14		9	Д9	
15		10	Д10	
16		11	Д11	
17		12	Д12	
18		13	Д13	
19		14	Д14	
20		15	Д15	
21	Общий	—	*GND*	—
22	Знак результата	—	*N*	Выход
23	Признак нулевого результата	—	*Z*	
24	Признак переполнения	—	*V*	
25	Перенос	—	*C*	
26	Не используются	—	—	—
27		—	—	—
28		—	—	—
29		—	—	—
30	Строб микрокоманды	—	ВК	Вход
31	Завершение выполнения микрокоманды	—	Ф1	Выход
32	Строб данных по магистрали Д	—	В	Вход-выход
33	Конец приема данных	—	П	
34	Шина микрокоманд	0	МК0	Вход
35		1	МК1	
36		2	МК2	
37		3	МК3	
38		4	МК4	
39		5	МК5	
40		6	МК6	
41		11	МК11	
42	Напряжение питания	—	U_{CC}	—

аккумулятор *АКК* и блок регистров общего назначения *БРОН*; используют в качестве источников и приемников информации при выполнении микрокоманд;

регистр состояния *РС*; в *РС* заносятся признаки результата операции. Программно доступен, может участвовать в операциях АЛУ;

блок синхронизации *БСЗ*; осуществляет обработку и выдачу внешних и внутренних сигналов управления. Работа *БСЗ* начинается с отрицательного фронта сигнала *ВК*;

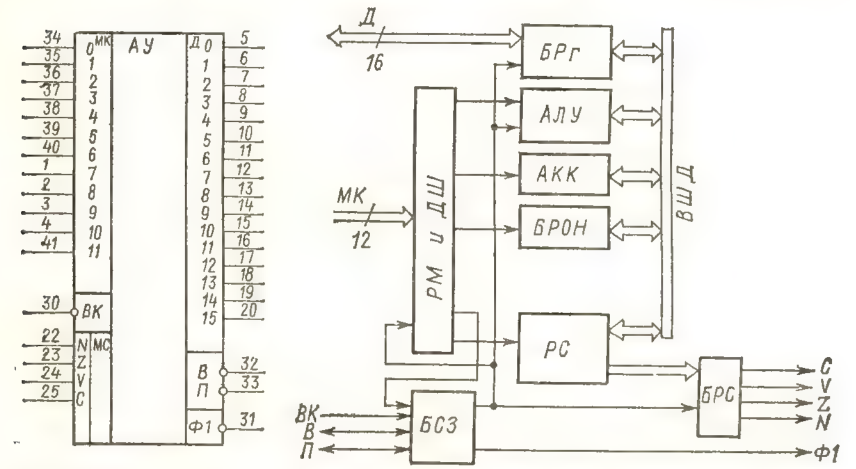


Рис. 6.4. Условное графическое обозначение микросхемы К588ВС2

Рис. 6.5. Структурная схема микросхемы К588ВС2

буферный регистр состояния *БРС*; через *БРС* происходит выдача признаков результата операции;

внутренняя шина данных *ВШД*; обеспечивает связь между *БРг, АЛУ, АКК, БРОН, РС.*

6.3.1. Описание функционирования микросхемы К588ВС2

Работа БИС АУ осуществляется под управлением микрокоманды. Выполнение любой микрокоманды состоит из комбинации фаз приема, чтения, записи и выдачи.

Фаза приема включает прием и дешифрацию микрокоманды. В некоторых микрокомандах в этой фазе происходит прием операнда по магистрали данных.

Фаза чтения включает чтение информации в *АЛУ* из *БРОН*, *АКК*, *РС* или *БРг* с последующей ее обработкой (рис. 6.6 ... 6.9).

В фазе записи происходят завершение обработки информации и ее запись в приемник информации. Приемником может быть

один из внутренних функциональных узлов БИС АУ — РОН, *АКК*, *РС* или *БРг* в соответствии с кодом микрокоманды.

В фазе выдачи осуществляется выдача результата операции в магистраль данных.

В табл. 6.5 ... 6.8 приведены микрокоманды, выполняемые БИС АУ.

Начало выполнения всех микрокоманд одинаковое. Если БИС АУ готова к приему микрокоманды, то на выходе *Ф1* (см.

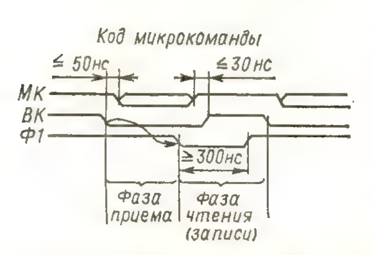


Рис. 6.6. Временна́я диаграмма работы микросхемы К588ВС2 при выполнении микропрограммы без обмена данными

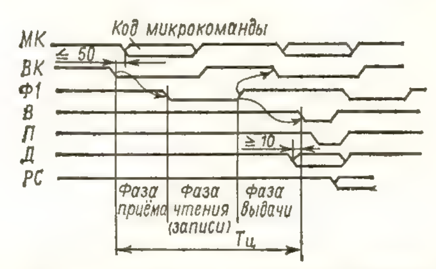


Рис. 6.7. Временна́я диаграмма работы микросхемы К588ВС2 при выполнении микрокоманды с выдачей информации

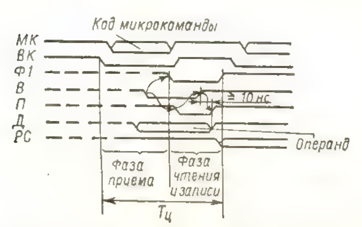


Рис. 6.8. Временна́я диаграмма работы микросхемы К588ВС2 при выполнении микрокоманды с приемом информации

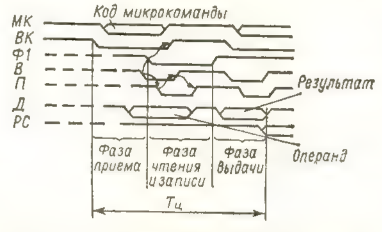


Рис. 6.9. Временна́я диаграмма работы микросхемы К588ВС2 при выполнении микрокоманды с приемом и выдачей информации

рис. 6.6 ... 6.9) устанавливается сигнал «1». После этого БИС УП (либо другое устройство микропрограммного управления) выдает микрокоманды на шину микрокоманд МК 12-разрядным кодом (0 ... 11); 12-й разряд шины микрокоманд, как правило, используется для стробирования микрокоманды и поступает на вход *ВК* сигналом «0». По отрицательному фронту этого сигнала происходит запись микрокоманды в регистр *РМ* и дешифратор *ДШ*. По окончании приема микрокоманды БИС АУ устанавливает сигнал «0» на выходе *Ф1*.

После этого БИС АУ переходит к исполнению микрокоманды.

Окончание выполнения микрокоманды инициируется положительным фронтом сигнала *Ф1*. Если микрокоманда выполняется без обмена, то не происходит ни приема операнда, ни выдачи результата операции в магистраль данных.

Таблица 6.5. Бинарные микрокоманды МК (0)=0

Разряды МК										Выполняемая операция
1	2	3	4	5	6	7	8	9	10	
1	0	КОП1: X	X	X	X (кроме X111)	РОН				АЛУ (РОН, АКК) → РОН
0	1	X	X	X	X (кроме X111)	РОН				АЛУ (РОН, АКК) → АКК
0	0	То же				РОН				АЛУ (РОН, АКК) → РС
1	1	То же				РОН				АЛУ (РОН, АКК) → АКК; БРг
0	0	X	1	1	1	X	X	X	X	Начальная установка
0	1	0	1	1	1	КОП2: X	X	X	X	АЛУ (БРг, АКК) → РС
1	1	0	1	1	1	X	X	X	X	АЛУ (БРг, АКК) → БРг, АКК
0	1	1	1	1	1	КОП2				АЛУ (РС, АКК) → АКК
1	0	1	1	1	1	КОП2				АЛУ (РС, АКК) → РС
1	1	1	1	1	1	КОП2				АЛУ (РС, АКК) → БРг, АКК

Выполнение микрокоманды с выдачей результата операции в магистраль данных обусловлена выдачей результата в магистраль совместно с формированием на выводе *В* сигнала «0». Одновременно с выдачей информации БИС АУ может принять следующую микрокоманду и начать ее выполнение. Этап выдачи информации в магистраль данных продолжается до поступления извне на вывод *П* БИС АУ отрицательного фронта сигнала. Последний инициирует положительный фронт сигнала *В* и снятие информации с магистрали данных. По положительному фронту сигнала *В* может сбрасываться сигнал *П*. На этом заканчивается выдача информации в магистраль. Если во время этапа выдачи информации в магистраль данных поступила микрокоманда без обмена, то последняя выполняется независимо от того, закончился или нет этап обмена информации в предыдущей микрокоманде. Если данная микрокоманда с приемом информации и еще не закончился этап выдачи данных в предыдущей микрокоманде, то БИС АУ не устанавливает уровень «0» на выходе *Ф1*, а ожидает появления «0» на выводе *П* (окончания этапа выдачи). Если следующая микрокоманда с выдачей и еще не закончился

Таблица 6.6. Унарные микрокоманды МК (0) = 1

Разряды МК 1 2	3 4 5 6	7 8 9 10	11	Выполняемая операция
0 X	КОП3 Любой, кроме: X 1 0 1; X 1 1 1		X	АЛУ (РОН) → РОН
0 X			X	БРг → РОН
1 0	0 1 0 1	РОН	0	МК (7 ... 10) → АКК (0 ... 3)
1 0			1	МК (7 ... 10) → АКК (4 ... 7)
1 1	0 1 0 1		0	МК (7 ... 10) → АКК (8 ... 11)
1 1			1	МК (7 ... 10) → АКК (12 ... 15)
0 1	1 1 1 1	КОП4	—	АЛУ (РС) → РС
1 X		Любой КОП4, кроме 1 1 0 1	—	АЛУ (РС) → БРг, РС
0 X	1 1 0 1		X	АЛУ (АКК) → АКК
1 X			X	АЛУ (АКК) → БРг, АКК
0 X	0 1 1 1	КОП4	X	АЛУ (БРг) → АКК
1 X			X	АЛУ (БП) → АКК; БРг

Таблица 6.7. Расшифровка КОП1 (КОП2)

Операция	КОП1 (2)	Действие	Признаки C	N	V	Z	Примечание
Конъюнкция	X 0 0 0	$A \wedge B \to$ ДД	—	+	0	+	—
Вычитание	X 0 0 1	$A - B \to$ ДД	+	+	+	+	—
Неэквивалент	X 0 1 0	$A \oplus B \to$ ДД	—	+	X	+	—
Сложение	0 0 1 1	$A + B \to$ ДД	+	+	+	+	—
Пересылка АКК	X 1 0 X	АКК → ДД	—	+	X	+	—
Сдвиг байта	1 0 1 1	A (0 ... 7) → ДД (8 ... 15)	—	+	0	+	МК (1,2) = 00
		A (8 ... 15) → ДД (0 ... 7)	—	+	0	+	МК (1,2) = 01
		A (0 ... 7) → ДД (8 ... 15)	—	+	0	+	МК (1,2) = 10
		A (8 ... 15) → ДД (0 ... 7)	—	+	0	+	МК (1,2) = 11
Дизъюнкция	X 1 1 0	$A \vee B \to$ ДД	—	+	0	+	—

Таблица 6.8. Расшифровка КОП3 (КОП4)

Операция	КОП3 (4)	Действие	Признаки			
			C	N	V	Z
Сдвиг логический влево	0 0 0 0	—	+	+	+	+
Сдвиг циклический влево	1 0 0 0	—	+	+	+	+
Декремент	0 0 0 1	$A - 1 \rightarrow ДД$	—	+	+	+
Дополнение	1 0 0 1	$A + 1 \rightarrow ДД$	+	+	+	+
Сложение с «C»	0 0 1 0	$A + C \rightarrow ДД$	+	+	+	+
Вычитание «C»	1 0 1 0	$A - C \rightarrow ДД$	+	+	+	+
Инкремент	0 0 1 1	$A + 1 \rightarrow ДД$	—	+	+	+
Сдвиг с «C» влево	1 0 1 1	—	+	+	+	+
Инверсия	0 1 0 0	$\overline{A} \rightarrow ДД$	+	+	0	+
Сдвиг с «C» вправо	1 1 0 0	—	+	+	0	+
Очистка	0 1 0 1	$(0) \rightarrow ДД$	0	0	0	+
Пересылка	1 1 0 1	$A \rightarrow ДД$	—	+	0	+
Сдвиг логический вправо	0 1 1 0	—	—	+	+	+
Сдвиг циклический вправо	1 1 1 0	—	+	+	+	+

этап выдачи информации в текущей микрокоманде, то после окончания выполнения фаз чтения и записи и выработки сигнала *Ф1* = «1» БИС АУ останавливается и ожидает сигнала *П* = «0». После появления этого сигнала сбрасывается сигнал *В* и после сброса сигнала *П* выдается новая информация в магистраль данных.

При выполнении микрокоманды с приемом информации по магистрали данных БИС АУ после приема и дешифрации кода этой микрокоманды переходит в режим ожидания приема операнда. Этап приема начинается с появления извне сигнала *В* = «0». Закончив прием, БИС АУ устанавливает сигнал *П* = «0» и сигнал *Ф1* = «0». После этого внешнее устройство (источник операнда) сбрасывает сигнал *В*, после чего БИС АУ сбрасывает сигнал *П*.

Окончание выполнения микрокоманды сигнализируется сигналом *Ф1* = «1». Микрокоманда с приемом операнда и выдачей результата операции аналогична микрокоманде с приемом информации из магистрали и фазой выдачи результата операции в магистраль данных.

6.3.2. Система микрокоманд БИС АУ

Система микрокоманд БИС АУ (см. табл. 6.5 ... 6.8) содержит бинарные (с двумя операндами) и унарные (с одним операндом) микрокоманды. При описании микрокоманд использованы следующие обозначения: МК — микрокоманда; РОН — регистр общего назначения; КОП — код операции; БРг — буферный регистр данных; АКК — аккумулятор; РС — регистр состояний; *А*, *В* — первый и второй операнды; *ДД* — приемник результата операции;

C, N, Z, V — признаки результата; X — безразличное состояние разряда.

Все микрокоманды можно условно разделить на следующие типы: без обмена информацией с магистралью данных; с приемом операнда по магистрали данных; с выдачей операнда в магистраль данных; с приемом и выдачей результата в магистраль данных.

В БИС АУ четыре формата микрокоманд. Бинарные и унарные микрокоманды имеют по два формата.

Во всех форматах нулевой разряд используется как идентификатор бинарной (МКО = «0») или унарной (МКО = «1») операций.

Одиннадцатый разряд является признаком выдачи информации состояния на магистраль состояния.

Для бинарных микрокоманд в форматах 1 и 2 разряды МК1 и МК2 определяют адрес приемника.

Для унарных микрокоманд разряд МК1 в форматах 3 и 4 определяет адрес приемника, а МК2 — указывает длину операнда (байт либо два байта).

Для форматов 1 и 3 разряды МК3 ... МК6 определяют код операций КОП1 и КОП3 соответственно.

Для форматов 2 и 4 эти разряды определяют признаки формата.

Для форматов 1 и 3 разряды МК7 ... МК10 в двоичном коде определяют номер РОНа.

Для форматов 2 и 4 эти разряды определяют код операций КОП2 и КОП4 соответственно.

Широкий спектр микрокоманд позволяет реализовать на базе БИС АУ практически любую систему команд вычислительного устройства.

Таблица 6.9. **Электрические параметры микросхемы К588ВС2 (КР588ВС2)**

Параметр, единица измерения	Обозначение	Значение
Напряжение питания, В	U_{CC}	5 ± 5 %
Статическая потребляемая мощность, мВт:	P_{CCS}	
для К588ВС2А (КР588ВС2А)		⩽1
для К588ВС2Б (КР588ВС2Б)		⩽5
Динамическая потребляемая мощность, мВт:	P_{CC}	
для К588ВС2А (КР588ВС2А)		⩽30
для К588ВС2Б (КР588ВС2Б)		⩽100
Входное напряжение «0», В	U_{IL}	$0 \leqslant U_{IL} \leqslant 0{,}4$
Входное напряжение «1», В	U_{IH}	$4{,}1 \leqslant U_{IH} \leqslant U_{CC}$
Входной ток «0», мкА	I_{IL}	⩽15
Входной ток «1», мкА	I_{IH}	⩽15
Выходной ток «0», мА	I_{OL}	⩾0,8
Выходной ток «1», мА	I_{OH}	⩾0,04
Время цикла при нормальных условиях, мкс	t_{CY}	1,82

На временных диаграммах (см. рис. 6.6 ... 6.9) иллюстрируется работа БИС АУ при выполнении различных микрокоманд. Индексы *В* и *П* указывают, что данный сигнал является выдаваемым либо принимаемым в БИС АУ.

В заключение следует отметить, что при проектировании устройств на базе БИС АУ необходимо разряды магистрали состояния «подвязывать» через резистор к источнику питания.

Величина резистора *R* определяется из формулы

$$\frac{5{,}5 - U_{OL}}{I_{OL}} \leqslant R \leqslant \frac{U_{CC} - U_{OH}}{I_{OH}}, \qquad (6.1)$$

где U_{CC}, U_{OL}, I_{OL}, U_{OH}, I_{OH} — см. в табл. 6.9.

В табл. 6.9 приведены электрические параметры БИС АУ.

6.4. МИКРОСХЕМА К588ВГ1 (КР588ВГ1)

Большая интегральная схема системного контроллера (БИС СК) предназначена для согласования и сопряжения процессора на базе микросхем серии К588 (КР588) с каналом типа *Q*-шина. Назначение выводов БИС СК дано в табл. 6.10. На рис. 6.10 и 6.11 изображены соответственно условное обозначение и структурная схема БИС СК.

В состав структурной схемы БИС СК входят следующие функциональные узлы: блок прерываний БП, дешифратор микрокоманд *ДШМ*, блок управления обменом данных *БУО*, блок управления прямым доступом к памяти *БУПД*, блок управления магистральными приемопередатчиками *БУПП*, блок синхронизации *БСЗ*.

Следует отметить, что линии управления обменом *ЛУО* включают следующие сигналы обмена (см. рис. 6.11): *ОБМ*, *ДЧТ*, *ДЗП*, *ПЗП*, *ОТВ*, *ЗПР*, *ПРР* и *ПВС*.

Линии передачи управления *ЛПУ* включают сигналы *ЗМ*, *РЗМ* и *ПЗ* (см. рис. 6.11).

БИС СК предназначена для связи внутреннего интерфейса процессора, построенного на базе микросхем серии К588 (КР588), с внешним стандартным интерфейсом канала *Q*-шина.

Внутренний интерфейс процессора содержит сигналы, обеспечивающие работу процессора, состоящего из магистральных приемопередатчиков, управляющей памяти и арифметического устройства. Это сигналы квитирования БИС АУ и БИС УП, 16-разрядная информационная магистраль, 4-разрядная магистраль кода прерывания, сигналы управления магистральными приемопередатчиками. Сигналы квитирования *КВ1* ... *КВ3* и *КП* обеспечивают передачу информации по 16-разрядной магистрали. Информация может включать команду, данные, адрес, вектор прерывания. Магистраль кода прерывания *ПР3* ... *ПР0* служит для сообщения в УП о том, какое прерывание в данный момент обслуживается.

Сигналы управления приемопередатчиками *С1* и *С2* предназначены для организации двунаправленного обмена информацией через МПП.

Внешний интерфейс процессора содержит сигналы, обеспечивающие работу всех устройств канала. Это информационные шины, шины управления вводом-выводом, шины прерываний и шины прямого доступа к памяти. Сигналы на шинах ввода-вывода

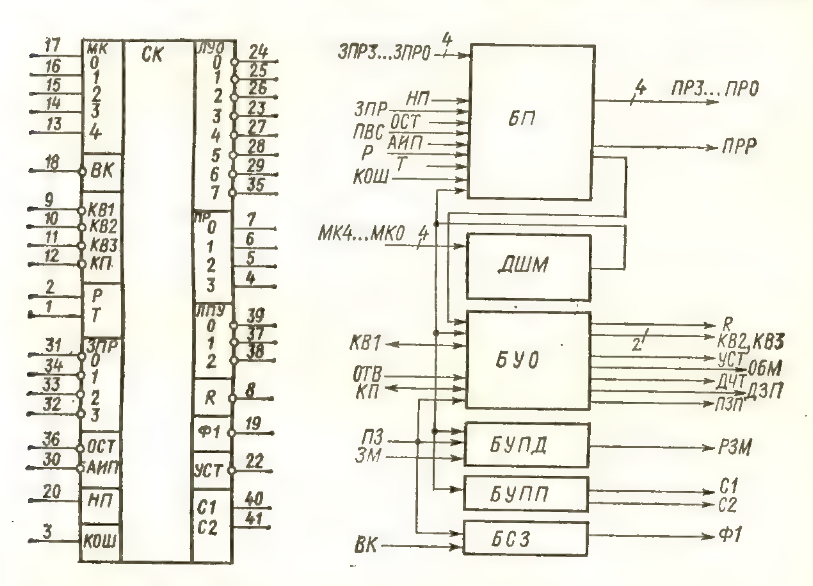


Рис. 6.10. Условное графическое обозначение микросхемы К588ВГ1

Рис. 6.11. Структурная схема микросхемы К588ВГ1

ОБМ, *ДЧТ*, *ДЗП*, *ПЗП*, *ОБМ* обеспечивают передачу по двунаправленной шине данных Д15 ... Д0 команд, адресов, данных и векторов прерываний.

Шины и сигналы прерываний *ЗПР3 ... ЗПР0*, *ОСТ*, *АИП*, *ЗПР*, *ПРР*, *ПВС* служат для приема запросов на прерывание и для сброса прерываний от внешних устройств.

Шины прямого доступа *ЗМ*, *РЗМ*, *ПЗ* служат для приема запросов на прямой доступ к памяти и сброса этих запросов.

Сигнал *УСТ* служит для приведения системы в исходное состояние, *КОШ* — вывод для подключения внешней времязадающей цепи для формирования длительности сигналов *УСТ*, *ДЧТ* или *ДЗП*.

БИС СК управляется микрокомандно. В табл. 6.11 приведена система микрокоманд БИС СК. Код микрокоманды поступает на входы МК4 ... МК0.

Таблица 6.10. **Назначение выводов микросхемы К588ВГ1**

Номер вывода	Назначение	Разряд	Обозначение	Тип
1 2	Регистр состояния	T P	T P	Вход
3	Подключение времязадающей цепи	—	КОШ	
4 5 6 7	Шина прерывания	3 2 1 0	ПР3 ПР2 ПР1 ПР0	Выход
8	Начальная установка	—	R	
9	Квитирование выдачи	1	КВ1	Вход-выход
10 11		2 3	КВ2 КВ3	Выход
12	Квитирование приема	—	КП	Вход-выход
13 14 15 16 17	Шина микрокоманд	4 3 2 1 0	МК4 МК3 МК2 МК1 МК0	Вход
18	Строб микрокоманды	—	ВК	
19	Квитирование приема микрокоманды	—	Ф1	Выход
20	Начальный пуск	—	НП	Вход
21	Общий	—	GND	—
22	Сброс канала	—	УСТ	Выход
23 24 25 26 27	Линии управления обменом	3 0 1 2 4	ЛУО3 ЛУО0 ЛУО1 ЛУО2 ЛУО4	
28		5	ЛУО5	Вход
29		6	ЛУО6	Выход
30	Внешний сигнал прерывания	—	АИП	Вход
31 32 33 34	Запрос прерывания с фиксированными векторами	0 3 2 1	ЗПР0 ЗПР3 ЗПР2 ЗПР1	
35	Линии управления обменом	7	ЛУО7	
36	Внешний сигнал прерывания	—	ОСТ	
37	Линии передачи управления	1	ЛПУ1	Выход
38 39		2 0	ЛПУ2 ЛПУ0	Вход
40 41	Управление приемопередатчиком	1 2	С1 С2	Выход
42	Напряжение питания	—	U_{CC}	—

Таблица 6.11. **Система микрокоманд микросхемы К588ВГ1**

Восьмеричный код	Обозначение	Назначение
00	INIT	Инициализация работы БИС СК
01	LPSW	Загрузка слова состояния программы
02	DATIV	Ввод вектора прерывания
03	WAIT	Ожидание
04	IE	Разрешение прерывания
05	ID	Запрет прерывания
10	DATI	Ввод слова
11	DATIK	Ввод команды
12	DATO	Вывод слова
13	DATOB	Вывод байта
14	DATIO	Ввод-вывод слова
15	DATIOB	Ввод-вывод байта
20	RTO	Сброс прерывания INTO
21	NH	Запрет прерывания HALT
22	RT1	Сброс прерывания INT1
23	RT2	Сброс прерывания INT2
24	RT3	Сброс прерывания INT3
25	RH	Сброс прерывания HALT
26	RTM	Сброс триггера маски
27	RES	Начальная установка
30	RTE	Сброс триггера ошибки
31	RTP	Сброс триггера РОК
32	CTO	Очистка триггера ТО
33	NTI	Запрет прерывания
34	REV	Сброс триггера
37	NOP	Нет операции

Рассмотрим выполнение трех микрокоманд, управляющих процессами ввода и вывода информации.

Микрокоманда ввода команды DATIK (рис. 6.12) выполняется следующим образом.

На вывод *КВ1* подается сигнал из БИС АУ о том, что на магистрали данных находится адрес команды. По этому сигналу БИС СК устанавливает сигналы разрешения обмена *ОБМ* = «0» и чтения данных *ДЧТ* = «0» для ОЗУ, а также разрешает БИС АУ снять с магистрали данных адрес команды, выдав сигнал *КП* = «0».

После снятия адреса команды БИС АУ выдает сигнал на вывод *КВ1* = «1», на что БИС СК отвечает сигналом *КП* = «1».

Затем на вывод *ОСТ* подается сигнал «0», свидетельствующий о том, что на магистрали данных находится команда и БИС СК разрешает УП принять команду. Кроме этого, при отсутствии запросов на прерывание БИС СК выдает сигналы *КВ2* = *КВ3* = = «0». Если к данному моменту запрос на прерывание поступит, то БИС СК вырабатывает сигнал *КВ3* = «0».

После приема команды БИС УП отвечает сигналом *КП* = «0», по которому БИС СК вырабатывает сигналы *КВ2* = *КВ3* = «1», а также сигнал *ДЧТ* = «1».

После снятия сигнала *ОСТ* (ОСТ = «1») БИС СК снимает сигнал *ОБМ* (*ОБМ* = «1»).

Выполнение микрокоманды ввода данных DATI (рис. 6.13) начинается с подачи из БИС АУ на БИС СК сигнала *КВ1* = «0», который свидетельствует о том, что адрес данных находится на магистрали.

По этому сигналу БИС СК устанавливает сигналы *ОБМ* = «0», *ДЧТ* = «0» (для ОЗУ или ВУ). Затем, после задержки относительно сигнала *ОБМ*, БИС СК устанавливает сигнал *КП* = «0», который разрешает снять адрес с магистрали данных. После этого АУ вырабатывает сигнал *КВ1* = «1», по которому БИС СК устанавливает сигнал *КП* = «1».
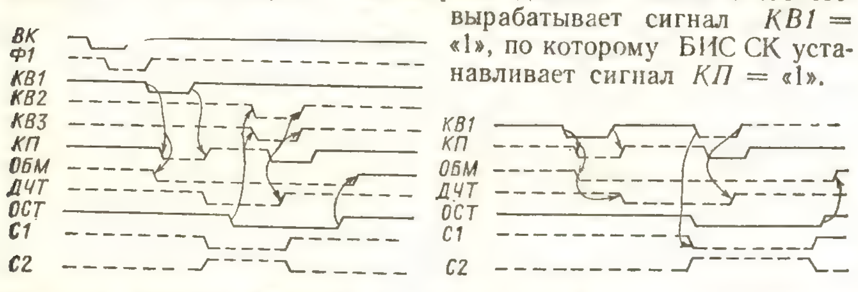


Рис. 6.12. Временна́я диаграмма выполнения микрокоманды DATIK

Рис. 6.13. Временна́я диаграмма выполнения микрокоманды DATI

На БИС СК подается сигнал *ОСТ* = «0» (данные находятся на магистрали). По этому сигналу БИС СК устанавливает сигнал *КВ1* = «0», разрешая БИС АУ принять данные. После окончания приема данных БИС АУ вырабатывает сигнал *КП* = «0». Вслед за этим БИС СК сбрасывает сигналы *КВ1* и *ДЧТ* в состояние «1». Получив сигнал *ОСТ* = «1», БИС СК вырабатывает сигнал *ОБМ* = «1».

Таблица 6.12. Коды на выводах ПР3 ... ПР0

Двоичный код	Мнемокод	Содержание прерываний
1 0 1 0	W (DBE)	Начальный пуск (двойная ошибка канала)
1 1 1 0	T (TERR0)	Прерывание по *T*-разряду
1 1 1 1	POK (TERR1)	Прерывание по сбою питания (ошибка канала)
0 1 1 1	HALT (TERR2)	Прерывание по входу ОСТ (ошибка канала)
0 1 0 1	INT0	Запрос прерывания 0 от ВУ
0 1 0 0	INT1	То же 1
0 1 1 0	INT2	» 2
0 0 1 0	INT3	» 3
0 0 1 1	EVNT	Прерывание по внешнему событию
0 0 0 1	IRQ	Запрос прерывания от ВУ с нефиксированным вводимым адресом-вектором
0 0 0 0	Нет прерывания	Нет прерывания

Выполнение микрокоманды вывода данных DATO (рис. 6.14) начинается с подачи из БИС АУ на БИС СК сигнала *КВ1* = «0», свидетельствующего о том, что на магистрали находится адрес данных. По этому сигналу БИС СК устанавливает сигналы *ОБМ* = *ПЗП* = «0» (для ОЗУ или ВУ). Затем с некоторой задержкой относительно сигнала *ОБМ* БИС СК устанавливает сигнал *КП* = «0». Последний разрешает БИС АУ снять адрес данных, после чего БИС АУ вырабатывает сигнал *КВ1* = «1». По этому сигналу БИС СК отвечает сигналами *ПЗП* = *КП* = «1».

АУ выставляет на магистраль данные, сигнализируя об этом сигналом *КВ1* = «0», по которому БИС СК устанавливает сигнал *ДЗП* = «0» (для ОЗУ или ВУ). После того, как данные запишутся в ОЗУ (ВУ) на БИС СК подается сигнал *ОСТ* = «0». По этому сигналу БИС СК вырабатывает сигналы *ДЗП* = «1» и *КП* = «0». Получив сигнал *КП* = «0», БИС АУ снимает данные с магистрали и устанавливает сигнал *КВ1* = «1», по которому БИС СК формирует сигнал *КП* = «1». По сигналу *ОСТ* = «1» БИС СК формирует сигнал *ОБМ* = «1».

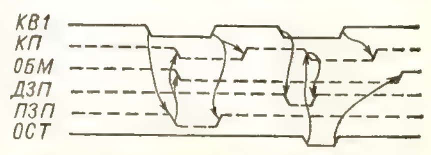


Рис. 6.14. Временна́я диаграмма выполнения микрокоманды DATO

В табл. 6.12 приведена расшифровка кодов, вырабатываемых БИС СК на выводах ПР3 ... ПР0.

Электрические параметры БИС СК приведены в табл. 6.13.

При разработке схемы с использованием БИС СК подключение времязадающей *RC*-цепочки к выводу КОШ осуществляется следующим образом: резистор *R* — между питанием и выводом КОШ; конденсатор *C* — между выводом КОШ и «минусом» источника питания. Значения резистора и конденсатора подбираются из условия обеспечения нужной длительности сигналов ДЧТ, ДЗП, до сигнала ОТВ.

Таблица 6.13. **Электрические параметры микросхемы К588ВГ1**

Параметр, единица измерения	Обозначение	Значение
Напряжение питания, В	U_{CC}	5±5 %
Статическая потребляемая мощность, мВт	P_{CCS}	$\leqslant 0{,}5$
Динамическая потребляемая мощность, мВт	P_{CC}	$\leqslant 15$
Входное напряжение «0», В	U_{IL}	$\leqslant 0{,}4$
Входное напряжение «1», В	U_{IH}	$4{,}1 \leqslant U_{IH} \leqslant U_{CC}$
Ток утечки по входу, мкА	I_{LIL} (I_{LIH})	$\leqslant 1{,}0$
Выходной ток «0», мА	I_{OL}	$\geqslant 0{,}8$
Выходной ток «1», мА	I_{OH}	$\geqslant 0{,}04$
Время выдачи адреса, нс	t_{BA}	$\leqslant 150$

Представляет собой БИС умножителя (БИС УМ) и выполняет операции умножения над 16-разрядными операндами в дополнительном коде.

На рис. 6.15 и 6.16 приведены условное графическое обозначение и структурная схема БИС УМ. Назначение выводов БИС УМ дано в табл. 6.14.

В состав структурной схемы БИС УМ входят следующие функциональные узлы:

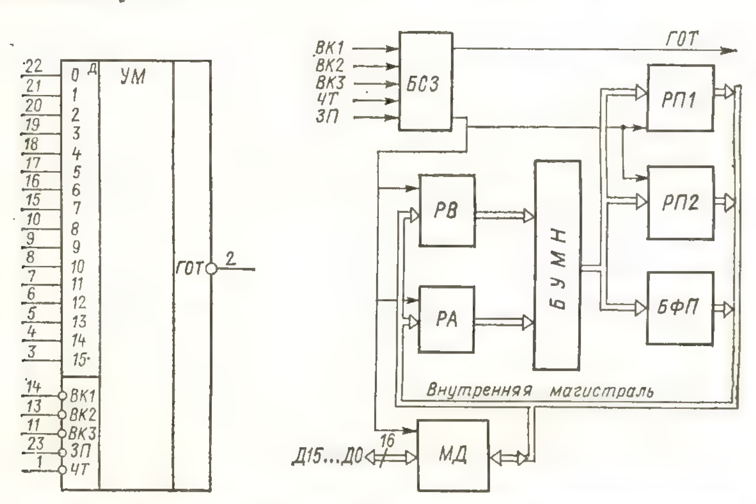


Рис. 6.15. Условное графическое обозначение микросхемы К588ВР2

Рис. 6.16. Структурная схема микросхемы К588ВР2

блок синхронизации *БСЗ*; предназначены для приема и обработки управляющих сигналов. *БСЗ* служит также для организации цикла выполнения операции умножения;

регистры операндов *А* и *В* — *РА* и *РВ*; служат для приема, хранения и выдачи в блок умножения множимого и множителя;

регистры произведения *РП1* и *РП2*; предназначены для приема из блока умножения БИС УМ и выдачи младшей и старшей частей произведения соответственно;

блок формирования признаков *БФП*; служит для формирования и выдачи признаков результата операции умножения. К ним относятся признак нулевого результата (*Z*), признак переполнения (*V*), признак переноса (*С*), знаковый разряд (*N*).

блок умножения *БУМН*; выполняет собственно операцию умножения операндов из регистров *РА* и *РВ*;

схема обмена информацией по магистрали данных *МД*; предназначена для обеспечения электрической «развязки» внутренней

магистрали БИС УМ с внешней магистралью данных, обеспечивает двунаправленный обмен 16-разрядной информацией.

Функционирование БИС УМН определяется сигналами *ВК1*, *ВК2*, *ВК3*, *ЧТ* и *ЗП* (рис. 6.17).

В первую очередь необходимо загрузить операнды в регистры *РА* и *РВ*. Загрузка 16-разрядной информации с магистрали в *РА* происходит после подачи на входы *ВК1*, *ВК2*, *ВК3*, *ЗП*, *ЧТ* комбинации сигналов «0 1 1 0 1» при наличии уровня «1» на выводе *ГОТ*.

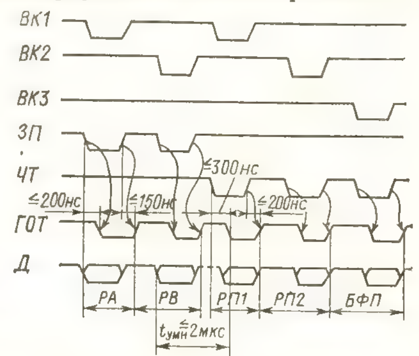


Рис. 6.17. Временная диаграмма работы микросхемы К588ВР2

Загрузка информации с магистрали данных в *РВ* происходит после подачи на указанные входы комбинации сигналов «1 0 1 0 1».

После загрузки второго операнда начинается выполнение собственно умножения. Младшая часть произведения считывается из *РП1* на магистраль данных после подачи на входы *ВК1*, *ВК2*, *ВК3*, *ЗП*, *ЧТ* комбинации сигналов «0 1 1 1 0».

Старшая часть произведения выдается из регистра *РП2* на магистраль данных при подаче на указанные входы комбинации сигналов «1 0 1 1 0». Если на те же входы подать комбинацию сигналов «1 1 0 1 0», то из блока *БФП* выдаются признаки результата умножения *N*, *Z*, *V*, *C* на магистраль данных.

В табл. 6.15 приведены электрические параметры БИС УМН.

Таблица 6.14. **Назначение выводов микросхемы К588ВР2**

Номер вывода	Назначение	Разряд	Обозначение	Тип
1	Выдача результата	—	ЧТ	Вход
2	Готовность	—	ГОТ	Выход
3	Магистраль данных	15	Д15	Вход-выход
4		14	Д14	
5		13	Д13	
6		12	Д12	
7		11	Д11	
8		10	Д10	
9		9	Д9	
10		8	Д8	
11	Строб	3	ВК3	Вход
12	Общий	—	*GND*	—
13	Строб	2	ВК2	Вход
14		1	ВК1	

Продолжение табл. 6.14

Номер вывода	Назначение	Разряд	Обозначение	Тип
15	Магистраль данных	7	Д7	Вход-выход
16		6	Д6	
17		5	Д5	
18		4	Д4	
19		3	Д3	
20		2	Д2	
21		1	Д1	
22		0	Д0	
23	Запись во внутренний регистр	—	ЗП	Вход
24	Напряжение питания	—	U_{CC}	—

Таблица 6.15. **Электрические параметры микросхемы К588ВР2**

Параметр, единица измерения	Обозначение	Значение
Напряжение питания, В	U_{CC}	5±5 %
Статическая потребляемая мощность, мВт	P_{CCS}	$\leqslant$5,0
Динамическая потребляемая мощность, мВт	P_{CC}	$\leqslant$35,0
Входное напряжение «0», В	U_{IL}	$0 \leqslant U_{IL} \leqslant 0{,}4$
Входное напряжение «1», В	U_{IH}	$4{,}1 \leqslant U_{IH} \leqslant U_{CC}$
Входной ток «0», и «1», мкА	I_{IL}, I_{IH}	$\leqslant$15
Выходной ток «0», мА	I_{OL}	$\geqslant$0,8
Выходной ток «1», мА	I_{OH}	$\geqslant$0,04
Время выполнения операции умножения, мкс	$t_{умн}$	$\leqslant$2,0

6.6. МИКРОСХЕМА К588ВА1 (КР588ВА1)

Представляет собой 8-разрядный магистральный приемопередатчик (МПП) и предназначен для построения устройств связи, контроллеров внешних устройств, интерфейсных блоков.

На рис. 6.18 и 6.19 изображены условное графическое обозначение и структурная схема МПП. Назначение выводов МПП приведено в табл. 6.16.

В состав структурной схемы МПП входят следующие функциональные узлы;

усилители каналов *К1* и *К2* (УК1, УК2); обеспечивают возможность работы МПП на 5 ТТЛ-нагрузок;

блок формирования контроля по четности *БФКЧ*; предназначен для формирования контрольных битов для информации, передаваемой по обоим каналам, а также для приема и обработки контрольных битов принимаемой информации. В случае появления ошибки по четности формируется сигнал *ОШ* = «0»;

блок управления передачами *БУП*; осуществляет управление передачей информации между каналами *К1* и *К2* в соответствии с сигналами *С1* и *С2*.

МПП осуществляет двунаправленный обмен информацией между каналами *К1* и *К2*. Обмен может быть осуществлен как с инвертированием передаваемой информации, так и без инверсии. Сигнал *ВП* = «0» сопровождает информацию, выдаваемую из МПП.

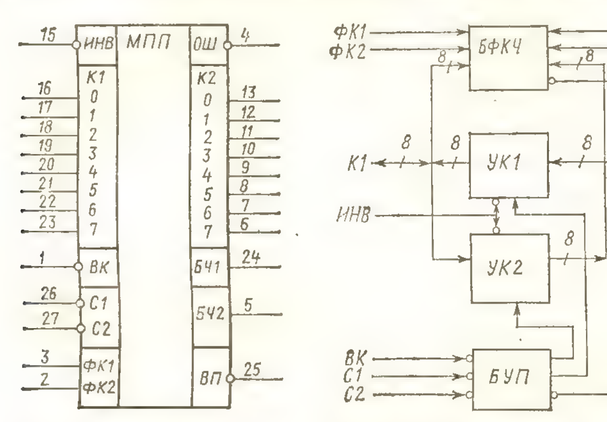


Рис. 6.18. Условное графическое обозначение микросхемы К588ВА1

Рис. 6.19. Структурная схема микросхемы К588ВА1

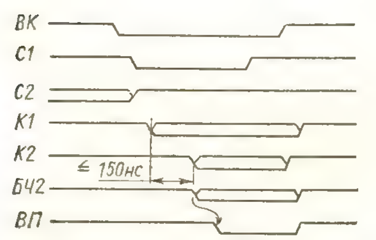


Рис. 6.20. Временна́я диаграмма работы микросхемы К588ВА1 при коммутации информации с канала К2 на К1

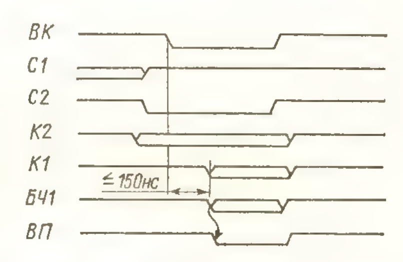


Рис. 6.21. Временна́я диаграмма работы микросхемы К588ВА1 информации с канала К1 на К2

Функционирование МПП начинается с подачи на вход *ВК* сигнала «0». При подаче на входы *С1* и *С2* комбинации сигналов «0» и «1» происходит передача информации из канала *К1* в канал *К2*. При подаче на эти же входы комбинации сигналов «1» и «0» происходит передача информации из канала *К2* в канал *К1*. При подаче на входы *С1* = *С2* = «1» выходы каналов *К1* и *К2* переводятся в третье состояние.

Подавать одновременно сигналы «0» на входы *С1* и *С2* запрещается.

Таблица 6.16. **Назначение выводов микросхемы К588ВА1**

Номер вывода	Назначение	Разряд	Обозначение	Тип
1	Выбор кристалла	—	ВК	Вход
2	Формирование—контроль четности канала К2	—	ФК2	Вход
3	Формирование—контроль четности канала К1		ФК1	Вход
4	Ошибка четности	—	ОШ	Выход
5	Бит четности канала К2	—	БЧ2	Вход-выход
6	Канал К2	7	К2-7	Вход-выход
7	Канал К2	6	К2-6	Вход-выход
8	Канал К2	5	К2-5	Вход-выход
9	Канал К2	4	К2-4	Вход-выход
10	Канал К2	3	К2-3	Вход-выход
11	Канал К2	2	К2-2	Вход-выход
12	Канал К2	1	К2-1	Вход-выход
13	Канал К2	0	К2-0	Вход-выход
14	Общий	—	*GND*	—
15	Инверсия	—	ИНВ	Вход
16	Канал К1	0	К1-0	Вход-выход
17	Канал К1	1	К1-1	Вход-выход
18	Канал К1	2	К1-2	Вход-выход
19	Канал К1	3	К1-3	Вход-выход
20	Канал К1	4	К1-4	Вход-выход
21	Канал К1	5	К1-5	Вход-выход
22	Канал К1	6	К1-6	Вход-выход
23	Канал К1	7	К1-7	Вход-выход
24	Бит четности канала К1	—	БЧ1	Вход-выход
25	Выполнено	—	ВП	Выход
26	Управление	1	С1	Вход
27	Управление	2	С2	Вход
28	Напряжение питания	—	U_{CG}	—

Таблица 6.17. **Работа микросхемы К588ВА1 при контроле передач информации**

Входы		Контроль на четность принимаемой информации		Формирование бита четности		Действие сигналов
ФК1	ФК2	К1, БЧ1	К2, БЧ2	БЧ1	БЧ2	
0	0	+	+	—	—	Контроль каналов К1 и К2
0	1	+	—	+	—	Контроль канала К1
1	0	—	+	—	+	Контроль канала К2
1	1	—	—	+	+	Формирование битов четности информации по любому каналу

Таблица 6.18. **Электрические параметры микросхемы К588ВА1**

Параметр, единица измерения	Обозначение	Значение
Напряжение питания, В	U_{CG}	5±5 %
Входное напряжение «1», В	U_{IH}	$4{,}1 \leq U_{IH} \leq U_{CG}$
Входное напряжение «0», В	U_{IL}	$0 \leq U_{IL} \leq 0{,}4$
Выходное напряжение «1», В	U_{OH}	$4{,}1 \leq U_{OH} \leq U_{CG}$
Выходное напряжение «0», В	U_{OL}	$0 \leq U_{OL} \leq 0{,}4$
Входной ток «0» и «1», мкА	I_{IL} (I_{IH})	$\leq 5{,}0$
Выходной ток «0», мА	I_{OL}	$\geq 8{,}0$
Выходной ток «1», мА	I_{OH}	$\geq 0{,}04$
Время передачи информации, нс	t_D	≤ 150

При *ИНВ* = «0» происходит инверсная передача информации между каналами *К1* и *К2*, при *ИНВ* = «1» — прямая.

Работа МПП при приеме и передаче контрольных битов представлена в табл. 6.17.

На рис. 6.20 и 6.21 приведены временные диаграммы функционирования МПП при обмене информации между каналами *К1* и *К2*.

Для обеспечения нормальной работоспособности МПП между выводами *ОШ*, *ВП* и питанием следует устанавливать доопределяющие резисторы. Значения резисторов можно определить из выражения (6.1), где U_{OL}, I_{OL}, U_{CC}, U_{OH}, I_{OH} — см. в табл. 6.18.

6.7. МИКРОСХЕМА К588ИР1 (КР588ИР1)

Представляет 8-разрядный многофункциональный буферный регистр (МБР) и может быть использована для построения регистровых блоков, контроллеров внешних устройств, запоминающих и интерфейсных устройств.

Условное графическое обозначение и структурная схема МБР представлены соответственно на рис. 6.22 и 6.23. В табл. 6.19 приведено назначение выводов МБР.

В состав структурной схемы МБР входят следующие функциональные узлы:

блок контроля четности *БКЧ*; осуществляет контроль на четность поступившей информации. При несоблюдении условия четности поступившей информации по каналу *К1* вырабатывается сигнал *ОШ* = «0», при этом *БКЧ* блокирует запись информации во внутренний регистр *Рг* и сигнал *ЗВ* не формируется;

блок формирования бита четности *БФБЧ*; предназначен для генерации бита четности (сигнал *БЧ*) после обработки 8-разрядной информации, поступившей с выходов внутреннего регистра *Рг*;

внутренний регистр *Рг*; несет основную функциональную нагрузку в МБР. Запись в регистр осуществляется сигналом, вырабатываемым блоком *БУЗ*;

блок управления записью *БУЗ*; предназначен для формирования сигнала записи информации с канала *К1* в *Рг*, а также для выработки сигнала квитирования *ЗВ*. Блокируется сигналом, вырабатываемым *БКЧ* при ошибке по четности;

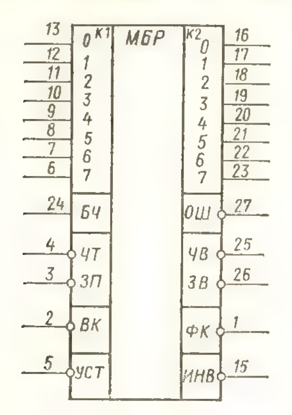


Рис. 6.22. Условное графическое обозначение микросхемы К588ИР1

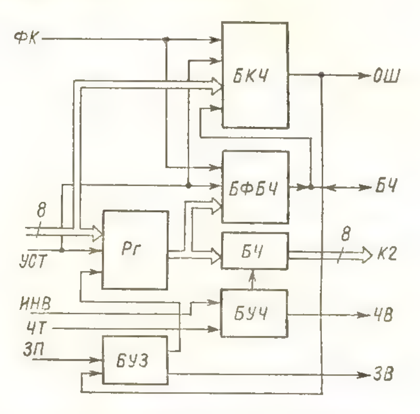


Рис. 6.23. Структурная схема микросхемы К588ИР1

буфер чтения *БЧ*; служит для приема информации с *Рг* и выдачи ее в канал *К2*;

блок управления чтением *БУЧ*; управляет работой *БЧ*, вырабатывая внутренний сигнал разрешения выдачи информации из *БЧ* в канал *К2*. Вырабатывает сигнал сопровождения информации, выдаваемой в канал *К2*.

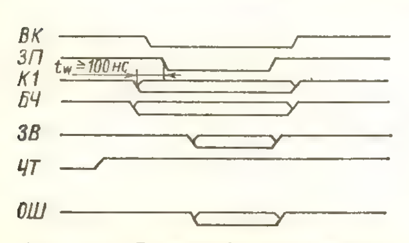


Рис. 6.24. Временна́я диаграмма работы микросхемы К588ИР1 в режиме «Запись»

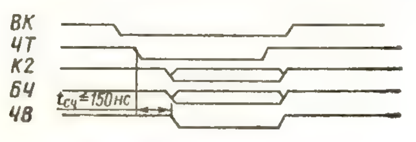


Рис. 6.25. Временна́я диаграмма работы микросхемы К588ИР1 в режиме «Чтение»

Функционирование МБР начинается с подачи сигнала «0» на вход *ВК*. При подаче сигнала «0» на вход ЧТ происходит выдача информации в канал *К2*. При подаче на вход *ЗП* отрицательного фронта сигнала происходит запись в МБР информации с канала *К2* (при отсутствии ошибки по четности). Если на входе *ЧТ* сигнал «1», то выходы канала *К2* находятся в третьем состоянии.

Сигнал *ЧВ* сообщает о наличии выдаваемой информации в канале К2, *ЗВ* — о завершении процесса записи информации в МБР из канала *К1*. Введение этих сигналов обусловлено использова-

Таблица 6.19. Назначение выводов микросхемы К588ИР1

Номер вывода	Назначение	Разряд	Обозначение	Тип
1	Формирование—контроль бита четности	—	ФК	Вход
2	Выбор кристалла	—	ВК	
3	Запись	—	ЗП	
4	Чтение	—	ЧТ	
5	Установка	—	УСТ	
6	Канал К1	7	К1-7	
7		6	К1-6	
8		5	К1-5	
9		4	К1-4	
10		3	К1-3	
11		2	К1-2	
12		1	К1-1	
13		0	К1-0	
14	Общий	—	*GND*	—
15	Инверсия	—	ИНВ	Вход
16	Канал К2	0	К2-0	Выход
17		1	К2-1	
18		2	К2-2	
19		3	К2-3	
20		4	К2-4	
21		5	К2-5	
22		6	К2-6	
23		7	К2-7	
24	Бит четности	—	БЧ	Вход-выход
25	Чтение выполнено	—	ЧВ	Выход
26	Запись выполнена	—	ЗВ	
27	Ошибка обмена	—	ОШ	
28	Напряжение питания	—	U_{CC}	—

Таблица 6.20. **Алгоритм функционирования микросхемы К588ИР1**

ВК	УСТ	ИНВ	ФК	ЧТ	ЗП	Операция
1	X	X	X	X	X	Нет операции
0	0	X	X	X	X	0 ⟶ МБР
0	1	1	1	0	1	(МБР) ⟶ К2, $\overrightarrow{БЧ}$
0	1	0	1	0	1	(МБР) ⟶ К2, $\overrightarrow{БЧ}$
0	1	X	0	1	0	К1 ⟶ МБР, $\overleftarrow{БЧ}$
0	1	1	1	0	0	К1 ⟶ МБР, (МБР) ⟶ К2, $\overrightarrow{БЧ}$
0	1	0	0	0	0	К1 ⟶ $\overline{МБР}$, (МБР) ⟶ К2, $\overleftarrow{БЧ}$

Примечание: (МБР) — содержимое внутреннего регистра МБР; → — направление пересылки информации, X — безразличное состояние сигналов; $\overrightarrow{БЧ}$, $\overleftarrow{БЧ}$ — сигнал БЧ, формируемый МБР и принимаемый извне.

нием общего принципа асинхронного обмена, который был применен при построении всех микросхем серии.

Сигнал *БЧ* может быть для *МБР* как входным, так и выходным. Если *ФК* = «0», то *БЧ* является входным для МБР и информация принимается по каналу *К1*. В этом случае МБР осуществляет «свертку» информации по модулю 2 и сравнение полученного результата «свертки» с входным сигналом *БЧ*. В случае совпадения осуществляется операция записи в МБР с выработкой выходного сигнала *ЗВ* = «0». При несовпадении указанных сигналов формируется сигнал *ОШ* = «0», запись в МБР блокируется и сигнал *ЗВ* = «0» не выдается.

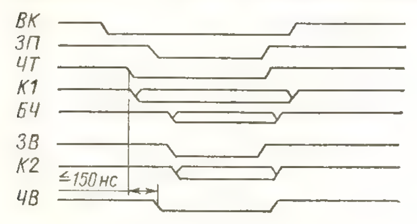


Рис. 6.26. Временна́я диаграмма работы микросхемы К588ИР1 в режиме «Чтение—Запись»

Если на входе *ФК* сигнал «1», то МБР формирует бит четности информации, выдаваемой по каналу *К2*.

Очистка регистра (установка в нуль) происходит при подаче на вход *УСТ* сигнала «0» независимо от сигналов на остальных входах МБР.

Вход *ИНВ* используется для осуществления выдачи информации инверсным кодом. В табл. 6.20 наглядно показано действие управляющих сигналов.

На рис. 6.24 ... 6.26 приведены временны́е диаграммы работы МБР, в табл. 6.21 — его электрические параметры.

При использовании микросхемы МБР необходимо между выводами *ОШ*, *ЧВ*, *ЗВ* и питанием подключать резисторы. Номинальные значения резисторов определяются так же, как и для микросхемы МПП (см. п. 6.3).

Таблица 6.21. **Электрические параметры микросхемы К588ИР1**

Параметр, единица измерения	Обозначение	Значение
Напряжение питания, В	U_{CC}	5±5 %
Входное напряжение «1», В	U_{IH}	$4{,}1 \leq U_{IH} \leq U_{CC}$
Входное напряжение «0», В	U_{IL}	$0 \leq U_{IL} \leq 0{,}4$
Выходное напряжение «1», В	U_{OH}	$4{,}1 \leq U_{OH} \leq U_{CC}$
Выходное напряжение «0», В	U_{OL}	$0 \leq U_{OL} \leq 0{,}4$
Статическая потребляемая мощность, мВт	P_{CCS}	$\leq 0{,}3$
Динамическая потребляемая мощность, мВт	P_{CC}	$\leq 10{,}0$
Выходной ток «1», мА	I_{OH}	$\geq 0{,}04$
Время считывания данных из МБР, нс	$t_{сч}$	≤ 150
Время записи данных в МБР, нс	t_W	≤ 100

6.8. МИКРОСХЕМА К588ВГ2 (КР588ВГ2)

Микросхема является контроллером запоминающего устройства (КЗУ) и предназначена для создания ЗУ различной организации, совместимых с каналом микроЭВМ «Электроника-60».

На рис. 6.27 представлено условное графическое обозначение, а в табл. 6.22 — назначение выводов микросхемы КЗУ.

В состав структурной схемы КЗУ (рис. 6.28) входят следующие функциональные узлы:

регистр адреса модуля ЗУ *РА*; предназначен для приема и запоминания старших разрядов адреса ЗУ. Защелкивание инфор-

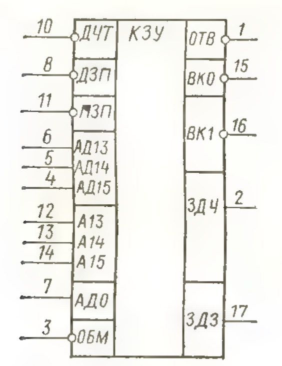


Рис. 6.27. Условное графическое обозначение микросхемы К588ВГ2

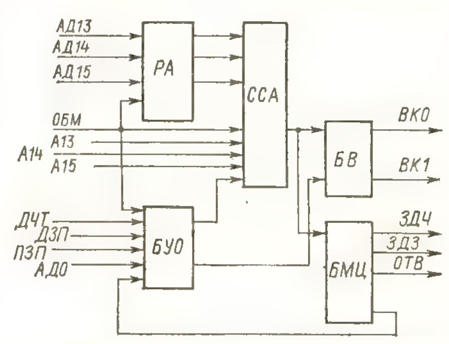


Рис. 6.28. Структурная схема микросхемы К588ВГ2

мации в *РА* происходит по отрицательному фронту сигнала *ОБМ*. Выходами подключен к первой группе входов *ССА*;

схема селекции адреса *ССА*; предназначена для сравнения комбинаций разрядов адреса *АД15* ... *АД13* с жестко заданным именным кодом модуля ЗУ (*А15* ... *А13*). Селектирование происходит при *ОБМ* = «0» и внутреннем сигнале разрешения селектирования. Последний вырабатывается блоком *БУО*;

блок управления обменом *БУО*; управляет работой блоков *БУО* и *БВ*;

блок выборки ЗУ *БВ*; предназначен для формирования сигналов *ВК0* и *ВК1*;

блок моделирования цикла ЗУ *БМЦ*; служит для настройки КЗУ на взаимодействие с блоками ЗУ, имеющими различные времена выборки. Настройка (моделирование) осуществляется установкой внешних времязадающих *RC*-цепочек. Последние подключаются к выводам *ЗДЧ* и *ЗДЗ*. Вырабатывает сигнал *ОТВ* = «0» по положительному фронту сигнала *ЗДЧ* и отрицательному фронту сигнала *ДЧТ*.

Функционирование КЗУ определяется управляющими сигналами *ОБМ*, *ПЗП*, *АДО*, ДЧТ и ДЗП (табл. 6.23)

При выполнении цикла чтения данных из модуля ЗУ по сигналу *ОБМ* = «0» происходит защелкивание старших разрядов адреса *АД13* ... *АД15* и сравнение их с заданным кодом на выводах *А13* ... *А15* (рис. 6.29). При совпадении этих кодов на выводе *ЗДЧ* появляется сигнал «0» и формируются сигналы *ВК0* = *ВК1* = = «0». На выводе *ЗДЧ* появляется положительный фронт, длительность которого определяется внешней *RC*-цепочкой. После этого по приходу сигнала *ДЧТ* = «0» КЗУ формирует сигнал *ОТВ* = «0». При переходе сигнала *ДЧТ* в состояние «1» прекращается выборка модуля ЗУ (*ВК0* = = *ВК1* = «1») и *ОТВ* переходит в состояние «1». Цикл завершается сигналом *ОБМ* = = «1». КЗУ готов принять следующую информацию.

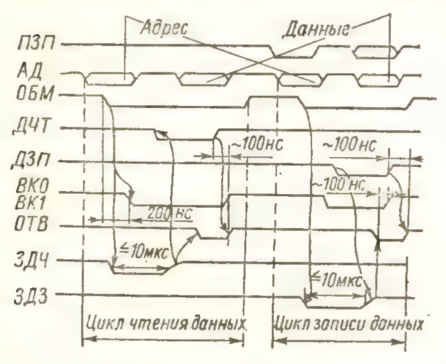


Рис. 6.29. Временная диаграмма работы микросхемы К588ВГ2

При выполнении цикла записи информации в модуль ЗУ так же осуществляется сравнение кодов *АД13* ... *АД15* с кодом на

Таблица 6.22. **Назначение выводов микросхемы К588ВГ2**

Номер вывода	Назначение	Разряд	Обозначение	Тип
1	Ответ устройства	—	ОТВ	Выход
2	Временное согласование по чтению	—	ЗДЧ	Выход
3	Синхронизация обмена	—	ОБМ	Вход
4	Шина адреса ЗУ	15	АД15	Вход
5	Шина адреса ЗУ	14	АД14	Вход
6	Шина адреса ЗУ	13	АД13	Вход
7	Шина адреса ЗУ	0	АД0	Вход
8	Управление записью 1	—	ДЗП	Вход
9	Общий	—	*GND*	—
10	Управление чтением	—	ДЧТ	Вход
11	Управление записью 2	—	ПЗП	Вход
12	Шина программирования адреса ЗУ	13	А13	Вход
13	Шина программирования адреса ЗУ	14	А14	Вход
14	Шина программирования адреса ЗУ	15	А15	Вход
15	Выборка младшего байта данных	—	ВК0	Выход
16	Выборка старшего байта данных	—	ВК1	Выход
17	Временное согласование с записью	—	ЗДЗ	Выход
18	Напряжение питания	—	U_{cc}	—

Таблица 6.23. **Операции микросхемы К588ВГ2**

Входные сигналы					Выходные сигналы		Операция
ОБМ	ДЧТ	ДЗП	ПЗП	АДО	ВК0	ВК1	
0	0	1	1	X	0	0	Чтение слова
0	1	0	1	X	0	0	Запись слова
0	1	0	0	0	0	1	Запись в младший байт
0	1	0	0	1	1	0	Запись в старший байт
1	X	X	X	X	1	1	Нет выборки

выводах *А13 ... А15*. По сигналу *ДЗП* = «0» формируются сигналы *ВК0* и *ВК1*. Напряжение на выводе ЗДЗ начинает расти (фронт определяется внешней *RC*-цепочкой) и вызывает формирование сигнала *ОТВ* = «0». Вслед за этим формируется положительный фронт сигнала *ДЗП*. Цикл заканчивается сигналом *ОБМ* =«1».

Следует отметить, что времязадающая *RC*-цепь подключается между выводами *ЗДЧ* и *ЗДЗ* и питанием (резистор и конденсатор, на каждый вывод подключаются параллельно). Кроме того, для обеспечения нормального функционирования необходимо подключать резистор между питанием и выводом *ОТВ*. Значение резистора находится из выражения, приведенного для микросхемы МПП. *R* и *C* времязадающих цепочек подбираются из условия согласования временной диаграммы работы модуля ЗУ с работой микросхемы СК. В табл. 6.24 приведены электрические параметры микросхемы КЗУ.

Таблица 6.24. **Электрические параметры микросхемы К588ВГ2**

Параметр, единица измерения	Обозначение	Значение
Напряжение питания, В	U_{CC}	5±5 %
Статическая потребляемая мощность, мВт	P_{CCS}	⩽0,3
Динамическая потребляемая мощность, мВт	P_{CC}	⩽15,0
Входное напряжение «0», В	U_{IL}	$0 \leqslant U_{IL} \leqslant 0{,}4$
Входное напряжение «1», В	U_{IH}	$4{,}1 \leqslant U_{IH} \leqslant U_{CC}$
Входной ток, мкА	I_I	⩽5,0
Выходной ток «0», мА	I_{OL}	⩾0,8
Выходной ток «1», мА	I_{OH}	⩾0,04
Время задержки выбора ЗУ, нс	$t_{ЗУ}$	⩽200

6.9. МИКРОСХЕМА К588ВТ1 (КР588ВТ1)

Представляет селектор адреса (СА) и предназначен для выполнения функций селектирования одного из восьми регистров внешних устройств, подключенных к *Q*-шине. Микросхема СА может

быть использована также для организации операций обмена содержимым внешних регистров как пословного, так и побайтного. Условное графическое обозначение СА представлено на рис. 6.30. В табл. 6.25 приведено назначение выводов СА. Структурная схема СА дана на рис. 6.31.

В состав структурной схемы СА входят следующие функциональные узлы:

регистр адреса *РА*; предназначен для буферизации поступающего 13-разрядного кода адреса. Старшая 9-разрядная его часть (*АД4 ... АД12*) с выходов *РА* поступает на *ССА*;

схема сравнения адреса *ССА*; предназначена для выполнения операции поразрядного сравнения кода, поступающего с выходов *РА*, с кодом *А4 ... А12*;

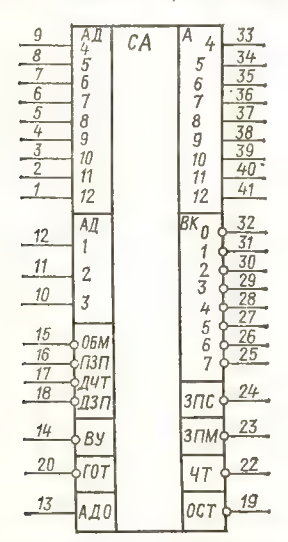


Рис. 6.30. Условное графическое обозначение микросхемы К588ВТ1

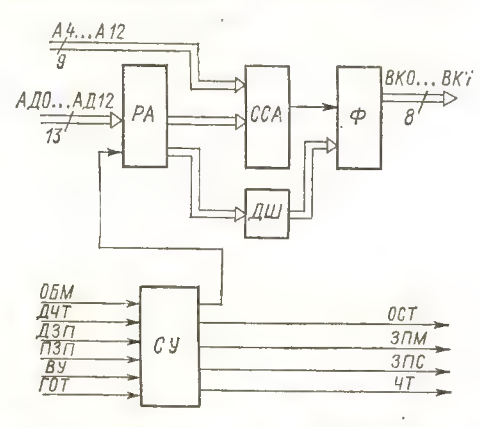


Рис. 6.31. Структурная схема микросхемы К588ВТ1

формирователь *Ф*; принимает выходную информацию с дешифратора *ДШ*; выдает сигналы *ВК0 ... ВК7*;

дешифратор *ДШ*; дешифрирует 3-разрядный код *АД1 ... АД3* при наличии сигналов *ОБМ* = *ВУ* = «0»;

схема управления *СУ*; обеспечивает взаимодействие микросхемы *СА* с *Q*-шиной и внешними устройствами.

Функционирование *СА* сводится к выполнению пяти циклов (см. п. 6.4): DATI, DATO, DATOB, DATIO, DATIOB.

Выполнение DATI (ввода слова) начинается с установки (рис. 6.32) процессором *адреса* на шине АД0 .. АД12 и сигналов *ВУ* = *ОБМ* = «0». Микросхема СА сравнивает девять старших разрядов этого адреса с разрядами жестко «зашитыми» на выходах А4 ... А12. Затем происходит дешифрация разрядов АД1 ... АД3 и выборка одного из восьми регистров одним из сигналов на шине *ВК*.

Таблица 6.25. **Назначение выводов микросхемы К588ВТ1**

Номер вывода	Назначение	Разряд	Обозначение	Тип
1		12	АД12	
2		11	АД11	
3		10	АД10	
4		9	АД9	
5		8	АД8	
6		7	АД7	
7	Шина адрес-данные	6	АД6	
8		5	АД5	
9		4	АД4	
10		3	АД3	Вход
11		2	АД2	
12		1	АД1	
13		0	АД0	
14	Выбор устройства	—	ВУ	
15	Синхронизация обмена	—	ОБМ	
16	Управление записью 2	—	ПЗП	
17	Управление чтением	—	ДЧТ	
18	Управление записью 1	—	ДЗП	
19	Ответ внешнего устройства	—	ОСТ	Выход
20	Готовность внешнего устройства	—	ГОТ	Вход
21	Общий	—	*GND*	—
22	Чтение	—	Чт	
23	Запись младшего байта	—	ЗПМ	Выход
24	Запись старшего байта	—	ЗПС	
25		7	ВК7	
26		6	ВК6	
27			ВК5	Выход
28		4	ВК4	
29	Шина выборки регистров	3	ВК3	
30		2	ВК2	
31		1	ВК1	
32		0	ВК0	
33		4	А4	
34		5	А5	
35		6	А6	
36		7	А7	
37	Шина программирования адреса	8	А8	Вход
38		9	А9	
39		10	А10	
40		11	А11	
41		12	А12	
42	Напряжение питания	—	U_{cc}	—

Процессор снимает адрес и сигнал *ВУ* и формирует сигнал *ДЧТ* = «0», по которому СА выставляет сигнал *ЧТ* = «0». Последний инициирует выбранное внешнее устройство на выдачу данных. Выдаваемые данные сопровождаются сигналом *ГОТ* = «0», по которому СА формирует сигнал *ОСТ* = «0». Получив данные, процессор снимает сигнал *ДЧТ*, а СА — сигнал *ЧТ*. Затем происходит цепочное снятие сигналов *ГОТ*, *ОСТ*, *ОБМ* и *ВК*.

При выполнении цикла DATO (вывод слова) процессор выдает на шину адреса информацию АД0 ... АД12 и выдает сигналы *ВУ* = *ОБМ* = *ПЗП* = «0» (рис. 6.33). Затем так же как и для цикла DATI происходит сравнение адреса, дешифрация 3-разрядного кода и выдача соответствующего разряда шины *ВК*. Процессор снимает адрес, сигналы *ВУ* и *ПЗП* и устанавливает данные и сигнал *ДЗП* = «0», по которому СА формирует сигналы *ЗПС* = *ЗПМ* = «0», разрешающие выбранному устройству запись данных. В ответ на поступившие данные внешнее устройство вырабатывает сигнал *ГОТ* = «0», сообщающий о том, что данные приняты. Затем СА устанавливает сигнал *ОСТ* = «0». Процессор снимает данные и сигнал *ДЗП*, по которому внешнее устройство снимает сигнал *ГОТ*, а СА — сигнал *ОСТ*.

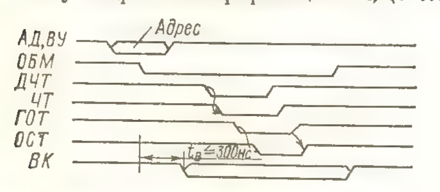


Рис. 6.32. Временна́я диаграмма работы микросхемы К588ВТ1 в цикле DATI

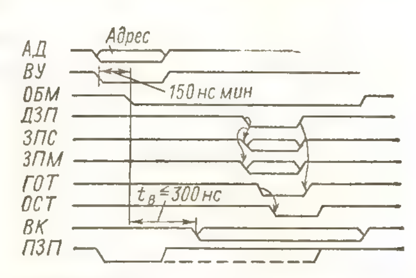


Рис. 6.33. Временна́я диаграмма работы микросхемы К588ВТ1 в цикле DATO

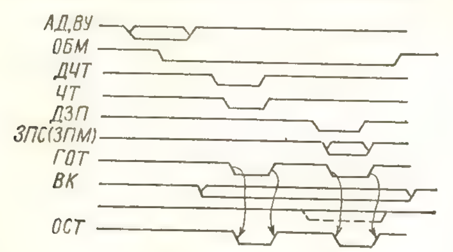


Рис. 6.34. Временна́я диаграмма работы микросхемы К588ВТ1 в цикле DATIO

Следующим действием является снятие процессором сигнала *ОБМ*, по которому СА снимает сигнал с шины *ВК*. Выполнение цикла DATOB (вывод байта) отличается от цикла DATO тем, что процессор сбрасывает сигнал *ПЗП* по снятию данных и в зависимости от значения разряда АД0 производятся действия над младшим (АД0 = «0») либо старшим (АД0 = «1») байтами слова. При этом вырабатываются сигналы *ЗПМ* либо *ЗПС* = «0».

Цикл DATIO (ввод — модификация — вывод слова) является смешанным. Первая его половина выполняется как в цикле DATI

Таблица 6.26. **Электрические параметры микросхемы К588ВТ1**

Параметр, единица измерения	Обозначение	Значение
Напряжение питания, В	U_{CG}	5±5 %
Статическая потребляемая мощность, мВт	P_{CGS}	≤2,5
Динамическая потребляемая мощность, мВт	P_{CG}	≤30,0
Входное напряжение «0», В	U_{IL}	$0 \leq U_{IL} \leq 0{,}4$
Входное напряжение «1», В	U_{IH}	$4{,}1 \leq U_{IH} \leq U_{CG}$
Входной ток, мкА	I_I	≤15,0
Выходной ток «0», мА	I_{OL}	≥0,80
Выходной ток «1», мА	I_{OH}	≥0,04
Время выборки, нс	$t_в$	≤300

(вплоть до выработки сигнала *ОСТ* = «0»). Вторая половина цикла выполняется как DATO, начиная с момента снятия процессором адреса. Однако в этом цикле процессор не устанавливает сигнал *ПЗП* = «0» (рис. 6.34).

Отличие выполнения цикла DATIOB (ввод — модификация — вывод байта) от DATIO в том, что при выводе данных процессор устанавливает сигнал *ПЗП* = «0».

На рис. 6.32 ... 6.34 приведены временные диаграммы выполнения циклов обмена.

В табл. 6.26 даны электрические параметры микросхемы СА.

6.10. ПРИМЕНЕНИЕ МИКРОСХЕМ СЕРИИ К588 (КР588)

На рис. 6.35 приведен пример использования микросхемы СА при совместной работе с БИС УМ. На рис. 6.36 дана структурная схема модуля ОЗУ, на которой показано включение микросхем КЗУ, МБР и МПП.

На рис. 6.37 приведена структурная схема процессора с системой команд микроЭВМ «Электроника-60». В состав процессора входят: устройство микропрограммного управления *УМУ*, операционный блок *ОБ* и распределитель информации *РИ*.

По магистрали данных поступает информация, в качестве которой могут быть переданы в (из) процессор(а) данные, адреса и команды. Код команды поступает на устройство *УМУ*. Здесь происходит преобразование команды в помикрокомандную последовательность. Разрядность микрокоманды процессора — 25. Микросхемы УП с «прошивками» 0001 ... 0004 предназначены для осуществления микропрограммного управления блоком *ОБ* (собственно АЛУ). Четыре БИС УП для управления *ОБ* необходимы прежде всего ввиду широкого набора команд «Электроники-60», а также в связи с ограниченными информационными возможно-

стями каждой БИС УП. БИС УП 0005 предназначена для выдачи микрокоманд разрешения прерывания, а также для управления блоком распределения информации. К сигналам разрешения прерывания ВУ относятся: ПРР0 (МК0) (с вектором 170); ПРР1 (МК1) (с вектором 174); ПРР2 (МК2) (с вектором 270); ПРР3 (МК3) (с вектором 274); ПРРВС (МК4) (по внешнему событию); ПРР ОСТ (МК5) (по «Останову»).

К сигналам управления обменом данных относятся: ОБМ — обмен данными; ДЧТ, ДЗП — чтение, запись данных; ОТВ —

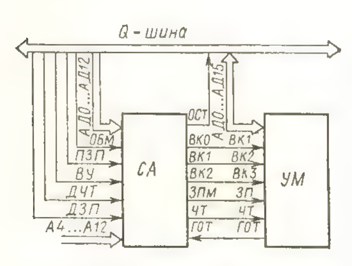


Рис. 6.35. Пример применения микросхемы К588ВР2 совместно с микросхемой К588ВТ1

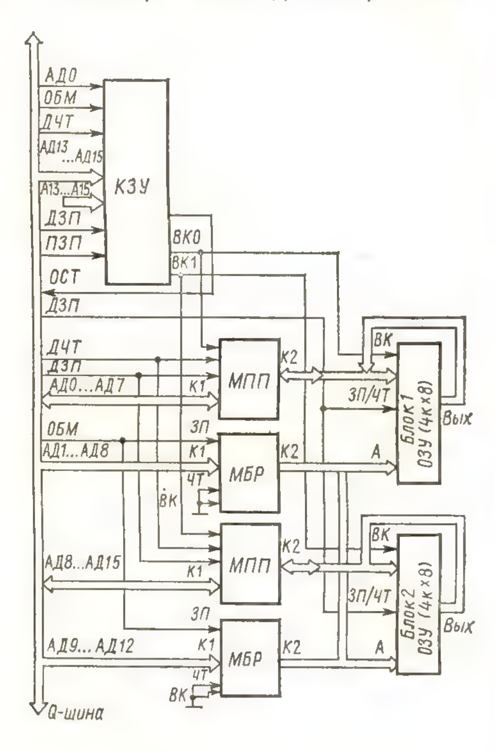


Рис. 6.36. Структурная схема ОЗУ

ответ; ПЗП — запись байта; ЗПР0 ... ЗПР3 — запросы прерывания ВУ с фиксированным вектором; ЗПР — запрос прерывания ВУ с вводимым вектором; ПРР — разрешение прерывания ВУ с вводимым вектором; ЗМ, РЗМ, ПЗ — сигналы обеспечения режима прямого доступа в память; ОСТ — аппаратный останов; АИП — авария источника питания; ПВС — прерывание по внешнему событию.

Разряды микрокоманды *МК7 ... МК11* управляют работой микросхемы СК. БИС СК обеспечивает выработку управляющих сигналов процессора *КВ1*, *КВ2*, которые инициируют прием командной (для БИС УП) либо операндной информации. В соответствии с этим прием информации происходит в операционный

блок, либо в *УМУ*. Тип обмена, который обеспечивают сигналы квитирования (ответа) информации *В*, *Ф1* — асинхронный. Ветвление (изменение последовательности выборки микрокоманд) происходит после анализа информации на магистрали *МС*.

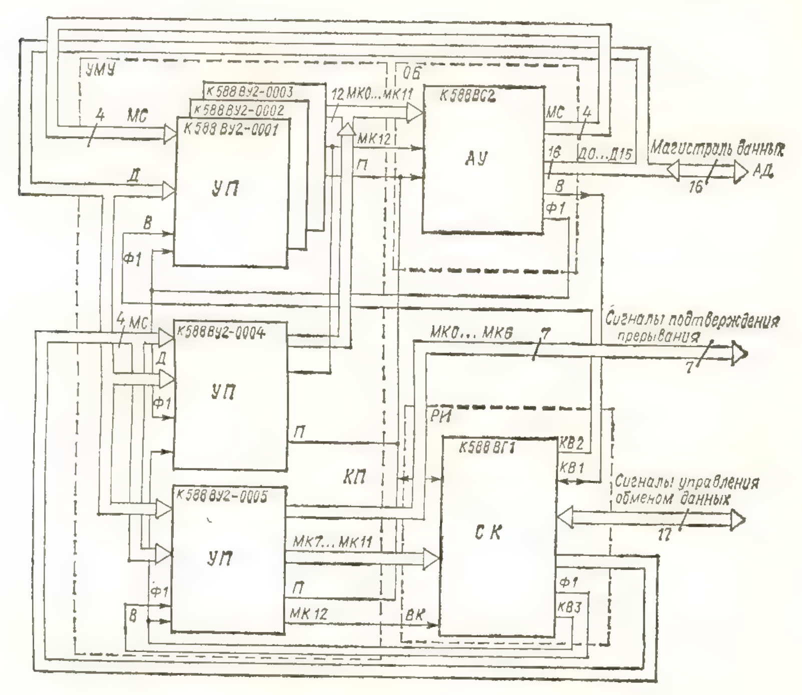


Рис. 6.37. Структурная схема процессора микроЭВМ

Глава 7

МИКРОПРОЦЕССОРНЫЙ КОМПЛЕКТ БИС СЕРИИ К1804

7.1. ОБЩИЕ СВЕДЕНИЯ

МПК БИС серии К1804 предназначен для создания блоков и узлов микроЭВМ, а также другой радиоэлектронной аппаратуры с высоким быстродействием. Благодаря модульности структуры, возможности параллельного наращивания, гибкости микропрограммного управления комплект обеспечивает построение устройств цифровой вычислительной техники различной архитектуры.

В состав серии включены микросхемы, реализующие функции арифметико-логических устройств, регистров общего назначения схем ускоренного переноса и управления адресами микрокоманд, интерфейсных схем, схем управления прерываниями и другие. Ряд схем комплекта служит для замены разнообразных по назначению схем обрамления, обычно реализуемых на микросхемах малой и средней степеней интеграции. В составе МПК планируется выпустить свыше 20 типов микросхем. Ниже рассмотрены первые шесть [2, 37, 42, 56]: микропроцессорная секция К1804ВС1; схема ускоренного переноса К1804ВР1; секции управления адресом микрокоманды К1804ВУ1 и К1804ВУ2; схема управления следующим адресом К1804ВУ3; параллельный регистр К1804ИР1.

Микросхемы серии К1804 выполнены по полупроводниковой биполярной технологии на основе транзисторно-транзисторной логики с диодами Шоттки (ТТЛШ) и совместимы с микросхемами ТТЛ серий К155, К555, К531 и другими, а также с МПК серий КР1802 и К589. Микросхемы комплектов серий К1804, КР1802 и К589 можно использовать совместно.

Наращиваемые секции комплекта 4-разрядные. Микросхемы серии К1804 выпускают в прямоугольных металлокерамических корпусах с вертикальным двусторонним расположением выводов (тип 2 по ГОСТ 17467-79). Основные характеристики микросхем приведены в табл. 7.1.

Электрические параметры микросхем приведены в соответствующих параграфах. Значения параметров даны (если это не оговорено особо) для всего диапазона температур (— 10 ... 70 °С). Напряжение питания микросхем серии К1804 5 В ± 5 %. Предельно допустимые значения электрических режимов эксплуатации микросхем приведены в табл. 7.2.

При одновременном воздействии повышенной температуры и электрического режима температура корпуса микросхемы не должна превышать 125 °С на основании корпуса или 120 °С на его крышке. Рекомендуемая температура корпусов не более 65 + 5 °С. Допустимое значение статического потенциала 30 В.

Свободные, не используемые в аппаратуре входы микросхем необходимо подключать к источнику постоянного напряжения

Таблица 7.1. **Основные характеристики микросхем серии К1804**

Тип микросхемы	Максимальное время задержки распространения сигнала, нс	Потребляемая мощность, мВт	Разрядность	Тип корпуса
К1804ВС1	100	1540	4	2123.40-6
К1804ВР1	19	600	4	201.16-13
К1804ВУ1	102	715	4	2121.28-6
К1804ВУ2	102	715	4	2140Ю.20-2
К1804ВУ3	60	630	—	201.16-13
К1804ИР1	21	715	4	201.16-13

Таблица 7.2. Предельно допустимые режимы эксплуатации микросхем серии К1804

Параметр, единица измерения	Обозначение	Значения параметров	
		мин.	макс.
Напряжение питания, В	U_{CC}	—	6,0
Напряжение на выходах, В	U_O	—	5,25
Емкость нагрузки, пФ	C_L	—	200
Напряжение на входах, В	U_I	—0,5	5,25
Выходной ток в состоянии низкого уровня, мА	I_{OL}	—	30
Длительность фронтов входных сигналов, нс	$t_ф$	—	100

(5,0 ± 0,25) В через резистор сопротивлением 1 кОм. К одному резистору можно подключать до 20 свободных входов.

Микросхемы можно прикреплять к печатной плате методом пайки выводов. Расстояние от корпуса до места пайки должно быть не ближе 1 мм. Лужение выводов в процессе монтажа микросхем следует производить в следующих режимах: температура припоя не более 250 °С; время нахождения выводов в расплавленном припое не более 2 с; интервал между двумя погружениями одних и тех же выводов не менее 5 мин.

7.2. МИКРОПРОЦЕССОРНАЯ СЕКЦИЯ К1804ВС1

Микропроцессорная секция (МПС) предназначена для построения операционных блоков цифровых ЭВМ. Гибкость команд МПС позволяет осуществлять эффективную эмуляцию различных ЭВМ.

МПС имеет:

микропрограммное управление;

возможность наращивания до любой разрядности, кратной четырем;

двухвходовое арифметико-логическое устройство (АЛУ), выполняющее 8 логических и арифметических операций и обладающее возможностями для организации ускоренного переноса;

регистровое запоминающее устройство (РЗУ) на 16 четырехразрядных слов с двумя независимыми каналами считывания;

дополнительный регистр с собственными цепями сдвига;

сдвигатель данных для результатов операций АЛУ при записи информации в РЗУ (операции АЛУ и сдвиг в одном цикле);

гибкий выбор источников и приемников данных АЛУ;

выработку в АЛУ четырех признаков результата: перенос, переполнение, знак, нуль;

трехстабильную шину вывода данных из МПС;

две пары двунаправленных шин сдвига.

Условное графическое обозначение МПС представлено на рис. 7.1. Назначение выводов МПС дано в табл. 7.3.

При описании БИС МПС и других микросхем комплекта используют положительную логику. Все обозначения сигналов на структурных схемах соответствуют обозначениям выводов в таблицах «Назначение выводов». При этом существует следующая связь обозначений выводов с обозначениями сигналов.

Если активному (разрешающему) состоянию на выводе соответствует напряжение высокого уровня, то обозначения выводов и сигналов на этих выводах в описании совпадают. В противном случае обозначение сигнала на выводе соответствует обозначению вывода с инверсией (например, $\overline{OE}$).

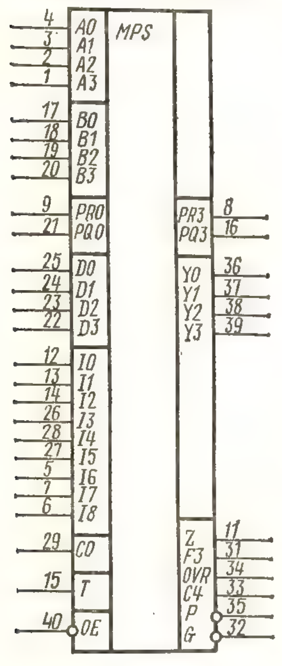


Рис. 7.1. Условное графическое обозначение МПС

Если информация на шинах в прямом коде, то обозначение сигналов на шинах совпадает с обозначением шин. В противном случае обозначение сигнала образуется как обозначение шины с инверсией.

Структурная схема МПС приведена на рис. 7.2. МПС включает следующие основные структурные единицы: блок внутренней памяти *БВП*, блок арифметико-логический *БАЛ*, блок регистра *PQ БР*, блок управления *БУ*.

В состав *БВП* входит регистровое запоминающее устройство *РЗУ* с двумя независимыми каналами выбора информации по адресам *A* и *B*. На выходах *РЗУ* включены два 4-разрядных регистра *PA* и *PB*, управляемых от тактового сигнала *T*. На входе *РЗУ* включен сдвигатель данных *АЛУ СДА*, позволяющий записывать в *РЗУ* информацию как без сдвига, так и со сдвигом вправо или влево на один разряд, причем запись в *РЗУ* возможна только по адресу *B*. *СДА* состоит из мультиплексора данных *АЛУ МДА* и двух буферов: буфера младшего разряда *АЛУ БМА* и буфера старшего разряда *АЛУ БСА*.

Блок арифметико-логический *БАЛ* содержит двухвходовое *АЛУ*, выполняющее три арифметические и пять логических операций и вырабатывающее четыре признака результата — перенос из старшего разряда *C*4, знак или содержимое старшего, третьего разряда на выходе *АЛУ* — *F*3, переполнение *OVR*, а также признак нулевого значения всех выходных шин *АЛУ* — *Z*. Для подключения схемы ускоренного переноса *АЛУ* вырабатывает сигналы генерации $\overline{G}$ и распространения $\overline{P}$ переноса. На входах *АЛУ* включен селектор источника данных *СИД*, позволяющий комбинировать в качестве пары операндов *АЛУ* информацию из

Таблица 7.3. **Назначение выводов микросхемы К1804ВС1**

Номер вывода	Назначение	Обозначение	Разряд	Тип
1	Адрес *A*	A3	3	Вход
2		A2	2	
3		A1	1	
4		A0	0	
5	Микрокоманда (выбор приемника)	I6	6	Вход
6		I8	8	
7		I7	7	
8	Вывод сдвига старшего разряда РЗУ	*PR*3	—	Двунаправленный
9	Вывод сдвига младшего разряда РЗУ	*PR*0	—	Двунаправленный
10	Питание	U_{CG}	—	—
11	Признак нулевого результата АЛУ	*Z*	—	Выход
12	Микрокоманда (выбор источника)	I0	0	Вход
13		I1	1	
14		I2	2	
15	Тактовый	*T*	—	Вход
16	Вывод сдвига старшего разряда регистра *PQ*	*PQ*3	—	Двунаправленный
17	Адрес *B*	B0	0	Вход
18		B1	1	
19		B2	2	
20		B3	3	
21	Вывод сдвига младшего разряда регистра *PQ*	*PQ*0	—	Двунаправленный
22	Данные	*D*3	3	Вход
23		*D*2	2	
24		*D*1	1	
25		*D*0	0	
26	Микрокоманда (выбор функции)	*I*3	3	Вход
27		*I*5	5	
28		*I*4	4	
29	Перенос АЛУ	*CO*	—	Вход
30	Общий	*GND*	—	—
31	Старший разряд АЛУ	*F*3	3	Выход
32	Сигнал генерации переноса АЛУ	*G*	—	Выход
33	Перенос АЛУ	*C*4	—	Выход
34	Переполнение АЛУ	*OVR*	—	Выход
35	Сигнал распространения переноса АЛУ	*P*	—	Выход
36	Данные	*Y*0	0	Выход
37		*Y*1	1	
38		*Y*2	2	
39		*Y*3	3	
40	Разрешение выходов	*OE*	—	Вход

Примечание. Выходы *Y*3 ... *Y*0, *PR*3, *PR*0, *PQ*3, *PQ*0 трехстабильные; выход *Z* с открытым коллектором.

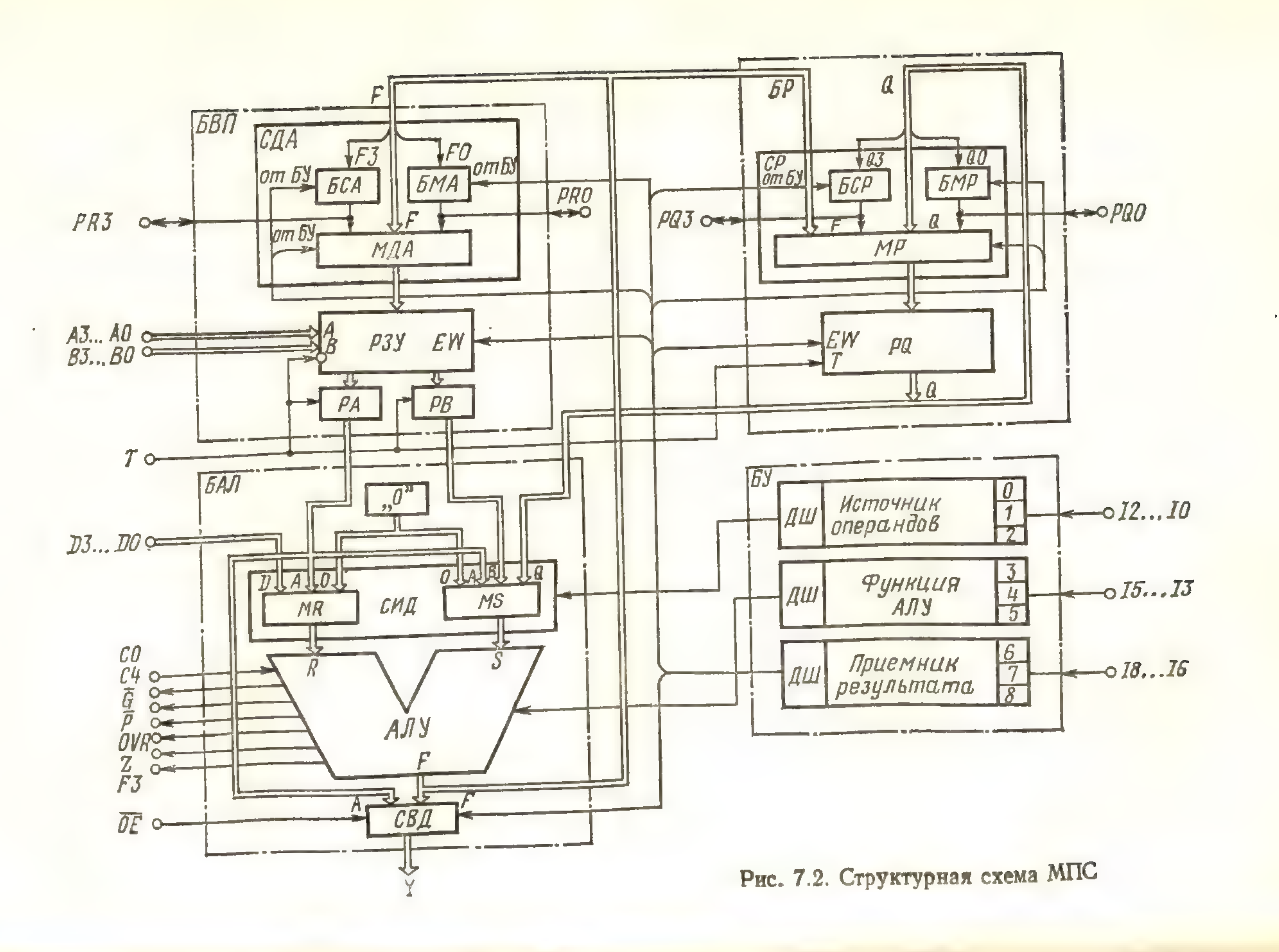


Рис. 7.2. Структурная схема МПС

пяти источников: с внешней шины данных *D*3 ... *D*0, с каналов *A* и *B* из *РЗУ* и из дополнительного регистра *PQ*. Пятый источник является условным и представляет собой нулевые значения, поданные на один из входов *АЛУ* (отсутствие информации на одном из входов). На структурной схеме МПС этот источник информации условно показан как блок «0». С выхода АЛУ информация подается на селектор выходных данных *СВД*. На второй вход селектора информация может быть передана прямо с выхода *A РЗУ*, минуя *АЛУ*. Информация через управляемые усилители, входящие в состав *СВД*, передается на выходную шину МПС — трехстабильную шину *Y*.

Блок регистра *БР* состоит из дополнительного регистра *PQ* и сдвигателя регистра *СР*, который позволяет переписывать информацию с выхода регистра *PQ* снова в него со сдвигом вправо или влево на один разряд. Данные из *АЛУ* заносятся в регистр *PQ* без сдвига.

Блок регистра *PQ* предназначен главным образом для выполнения умножения и деления, но может быть также использован в качестве аккумулятора и вспомогательного регистра, позволяющего выполнять операции со словами двойной длины.

Блок управления *БУ* формирует управляющие сигналы остальных блоков МПС. Входами в *БУ* являются шины кода микрокоманды *I*8 ... *I*0, которые, согласно формату микрокоманды, можно разделить на три группы. *БУ* также состоит из трех частей: первая вырабатывает сигналы управления для *СИД*; вторая — для *АЛУ*, а третья — для *СВД*, *РЗУ*, *PQ* и сдвигателей *СР* и *СДА*.

Рассмотрим подробно устройство и функционирование МПС.

На рис. 7.3 приведен фрагмент функциональной схемы МПС, представляющий *БВП*. Регистровое запоминающее устройство состоит из дешифраторов адреса A и B, шестнадцати 4-разрядных регистров общего назначения (РОН), схемы чтения и схемы записи. Дешифраторы адреса служат для выбора РОН, к которому производится обращение. Регистры общего назначения предназначены для хранения данных внутри МПС. Каждый из шестнадцати РОН может быть как источником информации, так и приемником результата операций. Выбор любого регистра в качестве источника информации осуществляется сигналами на адресных входах *A*3 ... *A*0 или *B*3 ... *B*0. Из *РЗУ* одновременно могут быть считаны два 4-разрядных слова, выбранные по входам A3 ... A0 и B3 ... B0, при этом, если на адресных входах установлены одинаковые адреса (обращение к одному и тому же РОН), на обоих выходах *РЗУ* (выходы *MA*3 ... *MA*0 и *MB*3 ... *MB*0; см. рис. 7.3) появляются идентичные данные, которые передаются на входы регистров. Считывание из *РЗУ* происходит при сигнале *T* высокого уровня («1»). Запись данных в любой из регистров *РЗУ* может производиться только в РОН, адресуемый сигналами на входах *B*3 ... *B*0, при этом запись осуществляется под управлением внешнего тактового сигнала на входе *T* и внутреннего сигнала разре-

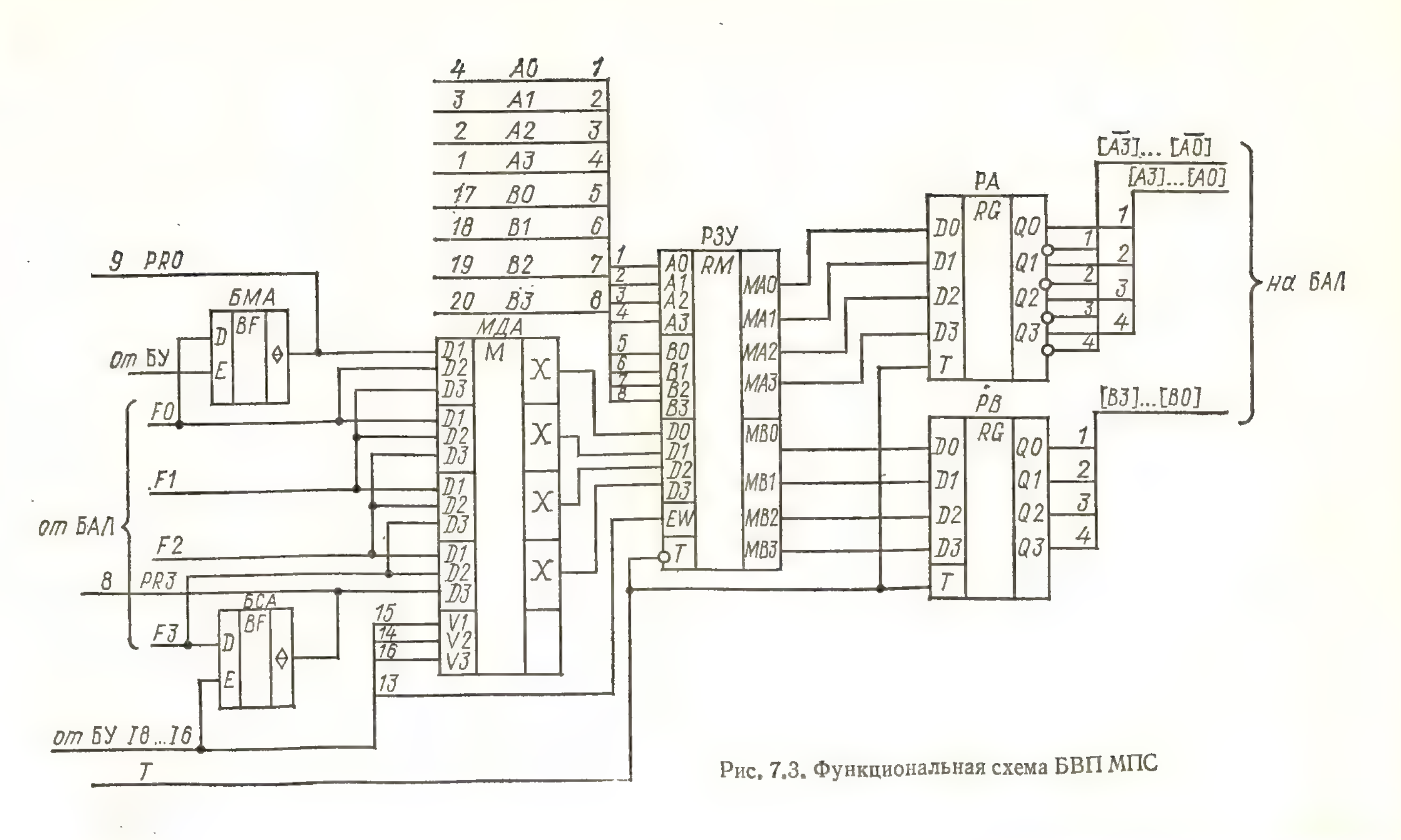


Рис. 7.3. Функциональная схема БВП МПС

шения записи *EW*, вырабатываемого блоком управления. Информация, поступающая на входы *РЗУ*, записывается в РОН, адресуемый сигналами на входах *В*3 ... *В*0 при наличии «1» на входе разрешения записи в *РЗУ* (*EW*) и «0» на тактовом входе *T*.

Регистры *РА* и *РВ*, принимающие данные *РЗУ* в соответствии с адресами *А* и *В* и передающие информацию в *БАЛ*, представляют собой 4-разрядные параллельные регистры с однофазной записью информации и однотактным управлением. Схема одного разряда регистра приведена на рис. 7.4. При единичном значении сигнала на входе синхронизации *С* на выход *Q* схемы передается информация, представленная на входе данных *D*. *Q* меняет свое состояние под воздействием сигнала на входе *D* до тех пор, пока на входе *С* остается «1». При нулевом значении сигнала на входе *С* на выходе *Q* схемы сохраняется состояние, которое было установлено до перехода сигнала на входе *С* с «1» на «0».

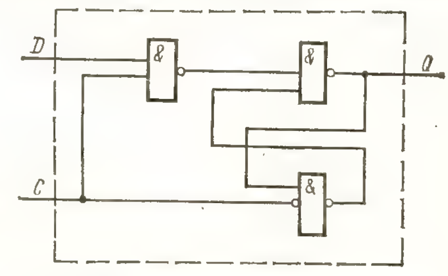


Рис. 7.4. Схема одного разряда регистров *А* и *В*

Таким образом, моментом начала записи по адресу *В*3 ... *В*0 является момент перехода тактового сигнала с уровня «1» к уровню «0». С этого момента информация на выходах *РА* и *РВ* фиксируется, т. е. сохраняется информация, введенная до изменения значения тактового импульса с «1» на «0». Это устраняет неопределенность в кольце передачи информации с выхода *БВП* на его вход. Все сигналы, управляющие работой МПС — сигналы кода микрокоманды, адресные сигналы, сигналы на двунаправленных шинах сдвига и другие — должны быть поданы до момента перехода тактового сигнала *T* с «1» в «0» с временным опережением, достаточным для установления переходных процессов в МПС.

После очередного перехода сигнала *T* из «0» в «1» начинается считывание информации из *РЗУ* по адресам *А* и *В* и эта информация появится на выходах *РА* и *РВ*.

Информация на входы *РЗУ* (см. рис. 7.2.) поступает со сдвигателя данных *АЛУ СДА*. Выводы *PR*0 и *PR*3 соединены с соответствующим входом *МДА*, включающего четыре трехвходовых мультиплексора (см. рис. 7.3), и соответствующим буфером *БМА*, *БСА*, имеющим трехстабильный выход. *СДА* управляется сигналами, формируемыми блоком управления при поступлении сигналов на входы *I*8 ... *I*6. Когда сдвиги не производятся, через входы *К*2 мультиплексоров (рис. 7.5, *а*) с выходов *АЛУ* передается информация, которая поступает на входы *РЗУ*. При этом *БМА* и *БСА* находятся в состоянии высокого сопротивления. В режиме сдвига в сторону старшего разряда (рис. 7.5, *б*) БМА имеет со-

стояние высокого сопротивления, $PR0$ становится входом, а $PR3$ — выходом *СДА* и через входы $K1$ мультиплексоров передается информация, сдвинутая на одну позицию, в результате чего на входы *РЗУ* поступает информация: $D0 = PR0$; $D1 = F0$; $D2 = F1$; $D3 = F2$, а содержимое старшего разряда *АЛУ* поступает на вывод $PR3 : PR3 = F3$. При сдвиге в сторону младших разрядов (рис. 7.5, *в*) *БСА* находится в состоянии высокого сопротивления и $PR3$ является входом сдвигателя, а $PR0$ — выходом. На вывод $PR0$ и на входы *РЗУ* передается: $PR0 = F0$; $D0 = F1$; $D1 = F2$; $D2 = F3$; $D3 = PR3$.

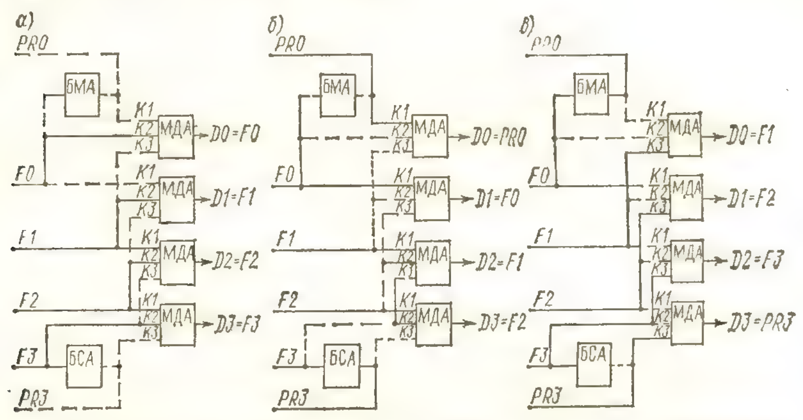


Рис. 7.5. Работа сдвигателя данных АЛУ: *а* — передача информации без сдвига; *б* — сдвиг в сторону старшего разряда; *в* — сдвиг в сторону младшего разряда

Блок регистра *PQ* включает регистр *PQ* и сдвигатель регистра *CP*. Структурная схема этого блока представлена в составе структурной схемы МПС (рис. 7.2), а на рис. 7.6 показан фрагмент функциональной схемы МПС, составляющий блок регистра *PQ*.

Регистр *PQ*, состоящий из триггеров *D*-типа, управляется тактовым импульсом. Запись информации в *PQ* производится по положительному фронту тактового импульса *T* при «1» на входе разрешения записи в регистр *EW*. *PQ* функционирует как аккумулятор промежуточных результатов при итеративных процессах вычисления. *PQ* может быть источником данных для *АЛУ* и приемником информации через *CP* с выходов *АЛУ* или собственных выходов.

Сдвигатель регистра *PQ* состоит из четырех трехвходовых мультиплексоров *MP*, входы которых подсоединены к выходам *АЛУ*, *PQ* и двунаправленным шинам $PQ0$ и $PQ3$, и двух буферов: буфера младшего разряда *PQ БМР* и буфера старшего разряда *БСР*. *CP* управляется сигналами, формируемыми блоком управления при поступлении сигналов на входы $I8 \dots I6$. При этом информация с выходов *АЛУ* передается на вход *PQ* без сдвига,

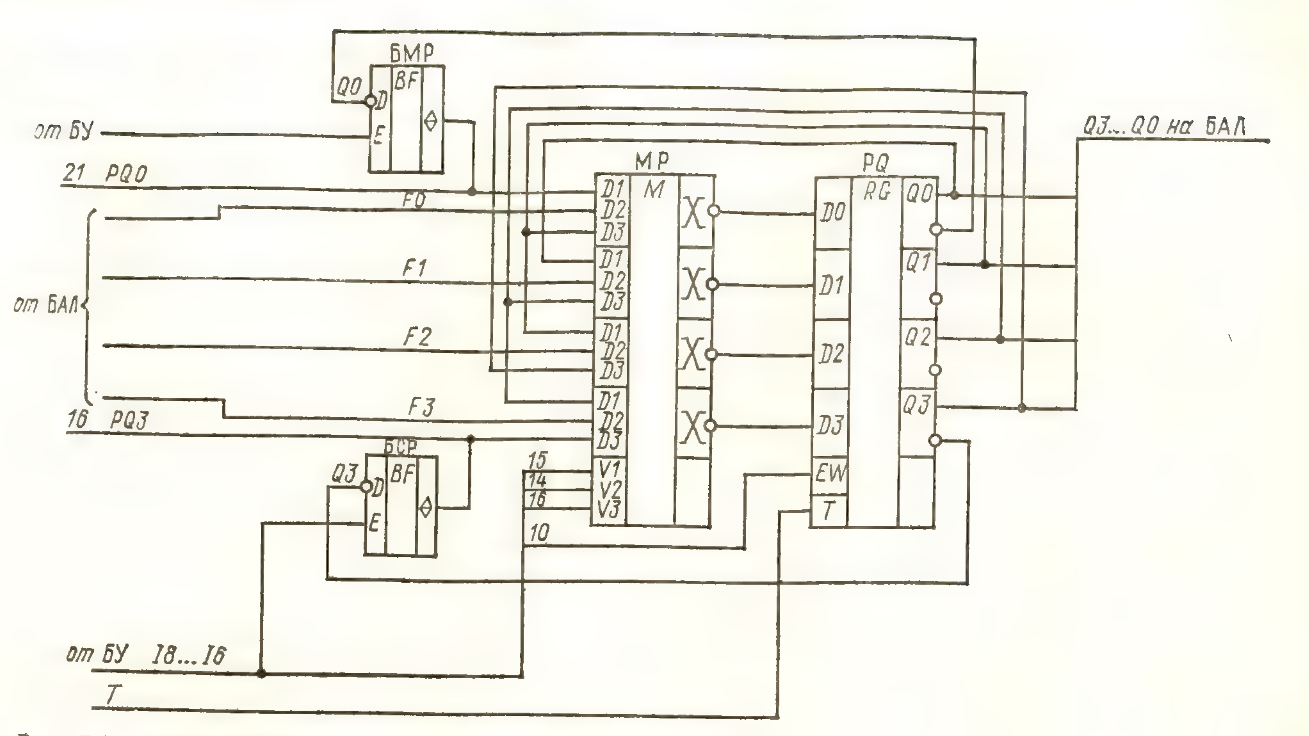


Рис. 7.6. Функциональная схема блока регистра

а информация с выхода *PQ* подается на вход *PQ* со сдвигом вправо или влево на один разряд. Работа *СР* осуществляется аналогично работе *СДА*, который был описан выше, а назначение выводов *PQ*0 и *PQ*3 подобно назначению выводов *PR*0 и *PR*3. Когда сдвигов не производится, на входы *D*2 мультиплексоров *MP* передается информация с выходов АЛУ (см. рис. 7.6) и на входы регистра *PQ* поступает информация: $D0 = F0$, $D1 = F1$, $D2 = F2$, $D3 = F3$. Буферы *БМР* и *БСР* имеют состояние высокого сопротивления (выводы *PQ*0 и *PQ*3 отключаются). При выполнении сдвигов на входы мультиплексоров *MP* поступает информация с выходов *PQ*. Одна из двунаправленных шин *PQ*0 или *PQ*3 при выполнении сдвигов используется в качестве входа *СР*, а другая отключается от *СР* и используется как выход буфера соответствующего разряда: *БМР*, *БСР*. При сдвиге в сторону старшего разряда (влево) *PQ*0 становится входом, *БМР* имеет состояние высокого сопротивления, на вывод *PQ*3 поступает старший разряд *PQ*, а через входы *D*1 мультиплексоров передается на входы *PQ*: $D0 = PQ0$, $D1 = Q0$, $D2 = Q1$, $D3 = Q2$. При сдвиге в сторону младшего разряда (вправо) входом становится *PQ*3 (*БСР* имеет состояние высокого сопротивления), на вывод *PQ*0 поступает младший разряд *PQ* и через входы *D*3 мультиплексоров в *PQ* передается информация: $D0 = Q1$, $D1 = Q2$, $D2 = Q3$, $D3 = PQ3$.

Блок арифметико-логический *БАЛ* (см. рис. 7.2) состоит из *АЛУ*, *СИД* и *СВД*. Фрагмент функциональной схемы МПС, составляющий арифметико-логический блок, приведен на рис. 7.7.

Быстродействующее *АЛУ* обеспечивает выполнение трех арифметических и пяти логических операций над одним или двумя четырехразрядными операндами (*R* и *S*). Информация на оба входа *АЛУ* поступает от *СИД*, который состоит из двухвходового мультиплексора входа *R АЛУ MR* и из трехвходового мультиплексора входа *S АЛУ MS*. СИД выбирает информацию на первый вход *АЛУ* (вход *R*) или от входов данных *D*3 ... *D*0 (D), или с выходов регистра *PA* блока внутренней памяти *БВП*. На вход *S АЛУ* может быть выбрана информация или с выхода регистра *PA БВП* (*A*), или с выхода регистра *PB БВП* (*B*), или с выхода регистра *PQ* (*Q*). Выбор источника данных *АЛУ* задается сигналами *I*2 ... *I*0. *СИД* обладает возможностью задерживать данные, что эквивалентно нулевому источнику операнда (блок «0»). Для этого на *СИД* подаются соответствующие сигналы управления с *БУ*, блокирующие выбор источника данных на соответствующий вход АЛУ. Очевидно, что при использовании двух источников возможны десять комбинаций пар источников операндов (*Q*0, *AB*, *AD*, *AQ*, *A*0, *BD*, *BQ*, *B*0, *DQ*, *D*0) и при одинаковых адресах *A* и *B* выбор *AD*, *AQ*, *A*0 и *BD*, *BQ*, *B*0 приводит к тождественным функциям. Таким образом, для *АЛУ* существует только семь полностью неизбыточных пар источников операндов. МПС использует восемь пар источников операндов, которые представлены в табл. 7.4. Из всех источников операндов *АЛУ* только сигналы

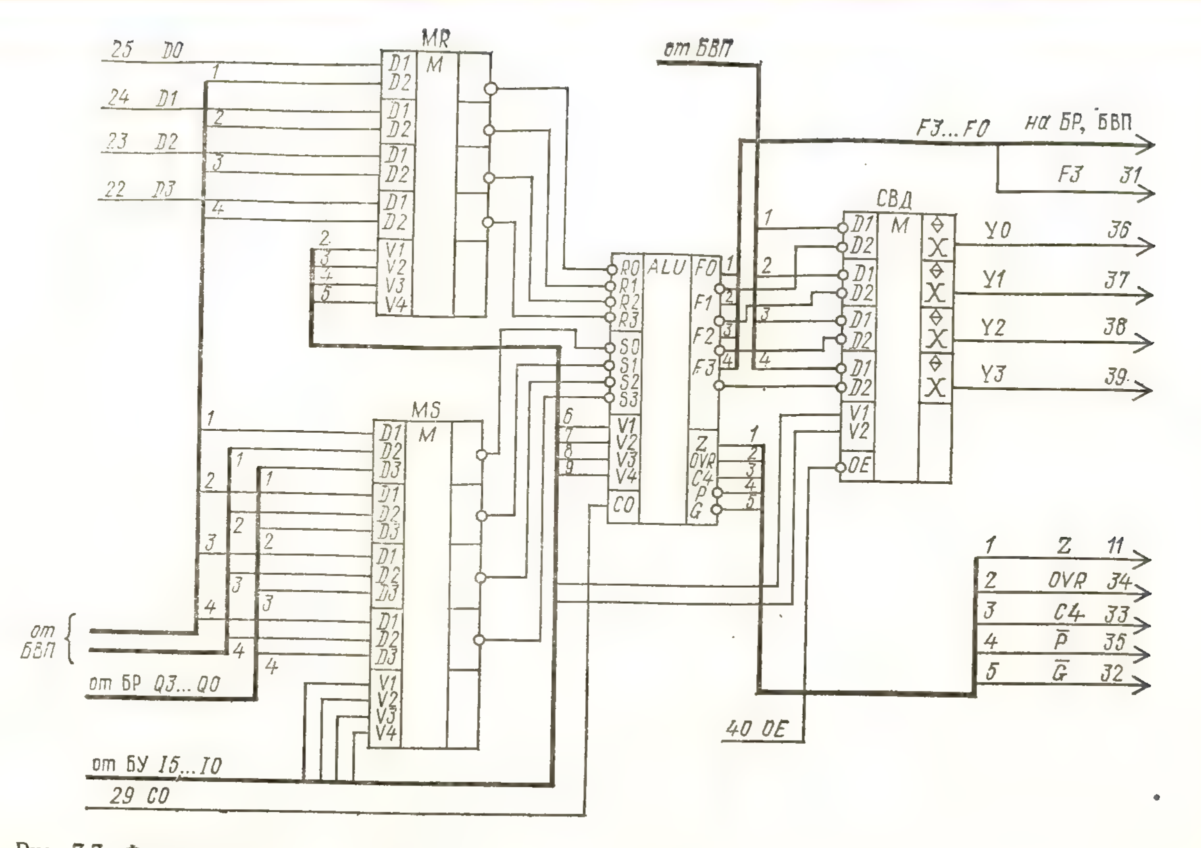


Рис. 7.7. Функциональная схема блока АЛУ

Таблица 7.4. **Управление источниками операндов АЛУ**

Микрокод				Источник операндов АЛУ	
*I*2	*I*1	*I*0	Восьмеричный код	*R*	*S*
0	0	0	0	*A*	*Q*
0	0	1	1	*A*	*B*
0	1	0	2	0	*Q*
0	1	1	3	0	*B*
1	0	0	4	0	*A*
1	0	1	5	*D*	*A*
1	1	0	6	*D*	*Q*
1	1	1	7	*D*	0

входов данных (*D*3 ... *D*0) являются входами МПС, остальные источники операндов — внутренние блоки МПС. Через входы *D*3 ... *D*0 МПС вводятся все данные, поступающие на МПС. Сигналы на входы данных могут поступать непосредственно из разрядов микрокоманды, либо от других источников информации. Выбор операции *АЛУ* осуществляется сигналами, которые формирует блок управления при поступлении на него сигналов со входов *I*5 ... *I*3 (табл. 7.5).

МПС (см. рис. 7.7) имеет выводы для приема в *АЛУ* входного переноса *C*0 и выдачи сигнала выходного переноса *C*4. Вход *C*0 и выход *C*4 предназначены для обеспечения нормального распространения последовательного переноса. При соединении нескольких МПС в качестве флажка переноса используется сигнал выходного переноса *C*4 самой старшей МПС. Реализация ускоренного переноса при увеличении длины слова осуществляется при использовании сигналов на выходах *АЛУ*: $\overline{G}$ — для переноса, образованного в *АЛУ* от данных, и $\overline{P}$ — для распространения переноса из младших разрядов.

Таблица 7.5. **Управление операциями АЛУ**

Микрокод				Операция АЛУ
*I*5	*I*4	*I*3	Восьмеричный код	
0	0	0	0	$R+S+C0$
0	0	1	1	$S-R-1+C0$
0	1	0	2	$R-S-1+C0$
0	1	1	3	$R \vee S$
1	0	0	4	$R \wedge S$
1	0	1	5	$\overline{R} \wedge S$
1	1	0	6	$R \oplus S$
1	1	1	7	$\overline{R \oplus S}$

Для представления чисел в *АЛУ* применяется дополнительный код. Поскольку МПС не имеет аппаратной реализации прибавления единицы, для образования дополнительного кода отрицательного числа путем прибавления единицы в младшем разряде используется вход переноса *С*0. Управление этим входом производится при всех арифметических операциях. Например, если оба операнда нулевые и выполняется вычитание, то при *С*0 = «0» на выходе *АЛУ* будет код 1111, обозначающий в дополнительном коде число минус 1. При *С*0 = «1» на выходе *АЛУ* будет код 0000, обозначающий в дополнительном коде нулевой результат. Логические операции *АЛУ* выполняются поразрядно. Описание операций *АЛУ* сведено в табл. 7.6 и 7.7.

Вывод *F*3 (см. рис. 7.7), старший разряд *АЛУ*, используется для определения знака результата операции *АЛУ*, независимо от того, в каком состоянии находится выходная шина *Y*. При соединении нескольких МПС знаковым является самый старший разряд. Единичное значение сигнала указывает, что число отрицательное.

Вывод переполнения (*OVR*) используется как флажок при арифметических операциях. Значение сигнала переполнения *OVR* определяется как «Исключающее ИЛИ» сигнала входного и выходного переноса старшего разряда *АЛУ*. Хотя каждая МПС вырабатывает сигнал *OVR*, при соединении нескольких МПС используется только сигнал переполнения самой старшей МПС, в остальных МПС сигналы переполнения игнорируются.

При нулевом значении всех выходов *F*, на выходе *Z* (с открытым коллектором) появляется «1». Для контроля нулевого результата операции *n*-разрядного *АЛУ* выводы *Z* от *n*-МПС можно объединить по схеме монтажное «ИЛИ».

Сигналы состояния $\overline{G}$, $\overline{P}$, *OVR*, *C*4 (см. рис. 7.7) имеют смысл при выполнении арифметических операций. Однако информация на этих выводах появляется и при выполнении логических операций. Логические правила образования этой информации приведены в табл. 7.8.

Информация с выхода *АЛУ* передается либо на *СВД*, либо на входы *СДА* и *СР* для выполнения соответствующих операций.

Селектор выходных данных управляет выводом информации из МПС. *СВД* представляет собой двухвходовые мультиплексоры, имеющие выходы с тремя состояниями, благодаря которым МПС может быть непосредственно подсоединена, например, к шинам микроЭВМ.

Через выходы *Y* МПС информация выводится при нулевом значении сигнала на входе *OE*, а при единичном значении этого сигнала вывод блокируется и выходы *Y* имеют состояние высокого сопротивления. Селектор выходных данных выбирает выход *АЛУ* или содержимое *БВП* по адресу *A* в зависимости от управляющих сигналов блока управления при подаче на него сигналов *I*8 ... *I*6.

Таблица 7.6. **Логические операции АЛУ**

Микрокод (восьмеричный код)		Группа	Операция
$I5$ $I4$ $I3$	$I2$ $I1$ $I0$		
4	0	И	$A \wedge Q$
4	1		$A \wedge B$
4	5		$D \wedge A$
4	6		$D \wedge Q$
3	0	ИЛИ	$A \vee Q$
3	1		$A \vee B$
3	5		$D \vee A$
3	6		$D \vee Q$
6	0	Исключающее ИЛИ	$A \oplus Q$
6	1		$A \oplus B$
6	5		$D \oplus A$
6	6		$D \oplus Q$
7	0	Исключающее ИЛИ—НЕ	$\overline{A \oplus Q}$
7	1		$\overline{A \otimes B}$
7	5		$\overline{D + A}$
7	6		$\overline{D + Q}$
7	2	Инвертирование	$\bar{Q}$
7	3		$\bar{B}$
7	4		$\bar{A}$
7	7		$\bar{D}$
6	2	Передача	Q
6	3		B
6	4		A
6	7		D
3	2	Передача	Q
3	3		B
3	4		A
3	7		D
4	2	Нуль	0
4	3		0
4	4		0
4	7		0
5	0	Маска	$\bar{A} \wedge Q$
5	1		$\bar{A} \wedge B$
5	5		$\bar{D} \wedge A$
5	6		$\bar{D} \wedge Q$

Таблица 7.7. **Арифметические операции АЛУ**

Микрокод (восьмеричный код)		$C0$ = «0»		$C0$ = «1»	
$I5 \ldots I3$	$I2 \ldots I0$	Группа	Операция	Группа	Операция
0	0	Сложение	$A+Q$	Сложение плюс один	$A+Q+1$
0	1		$A+B$		$A+B+1$
0	5		$D+A$		$D+A+1$
0	6		$D+Q$		$D+Q+1$
0	2	Передача	Q	Инкремент	$Q+1$
0	3		B		$B+1$
0	4		A		$A+1$
0	7		D		$D+1$
1	2	Декремент	$Q-1$	Передача	Q
1	3		$B-1$		B
1	4		$A-1$		A
2	7		$D-1$		D
2	2	Обратный код	$-Q-1$	Дополнительный код	$-Q$
2	3		$-B-1$		$-B$
2	4		$-A-1$		$-A$
1	7		$-D-1$		$-D$
1	0	Вычитание в обратном коде	$Q-A-1$	Вычитание в дополнительном коде	$Q-A$
1	1		$B-A-1$		$B-A$
1	5		$A-D-1$		$A-D$
1	6		$Q-D-1$		$Q-D$
2	0		$A-Q-1$		$A-Q$
2	1		$A-B-1$		$A-B$
2	5		$D-A-1$		$D-A$
2	6		$D-Q-1$		$D-Q$

Таблица 7.8. Формирование сигналов состояния при выполнении различных операций АЛУ

$I5 \ldots I3$	Функция	$\overline{P}$	$\overline{G}$	$C4$	OVR
0	$R+S$	$\overline{P3\cdot P2\cdot P1\cdot P0}$	$\overline{G3+P3\cdot G2+P3\cdot P2\cdot G1+P3\cdot P2\cdot P1\cdot G0}$	$C4$	$C3\oplus C4$
1	$S-R$	Как для $R+S$, но Ri меняется на $\overline{Ri}$			
2	$R-S$	Как для $R+S$, но Si меняется на $\overline{Si}$			
3	$R\vee S$	0	$P3\cdot P2\cdot P1\cdot P0$	$\overline{P3\cdot P2\cdot P1\cdot P0+C0}$	$\overline{P3\cdot P2\cdot P1\cdot P0+C0}$
4	$R\wedge S$	0	$\overline{G3+G2+G1+G0}$	$G3+G2+G1+G0+C0$	$G3+G2+G1+G0+C0$
5	$\overline{R}\wedge S$	0	Как для $R\wedge S$, но Ri меняется на $\overline{Ri}$		
6	$R\oplus S$	Как для $\overline{R\oplus S}$, но Ri меняется на $\overline{Ri}$			
7	$\overline{R\oplus S}$	$G3+G2+G1+G0$	$G3+P3\cdot G2+P3\cdot P2\cdot G1+P3\cdot P2\cdot P1\cdot P0$	$\overline{G3+P3\cdot G2+P3\cdot P2\cdot G1+P3\cdot P2\cdot P1\cdot P0\cdot(G0+\overline{C0})}$	OVR (7)

Примечания: 1. $P0 = R0 + S0$, $P1 = R1 + S1$, $P2 = R2 + S2$, $P3 = R3 + S3$, $G0 = R0\cdot S0$, $G1 = R1\cdot S1$, $G2 = R2\cdot S2$, $G3 = R3\cdot S3$, $C3 = G2 + P2\cdot G1 + P2\cdot P1\cdot G0 + P2\cdot P1\cdot P0\cdot C0$, $C4 = G3 + P3\cdot G2 + P3\cdot P2\cdot G1 + P3\cdot P2\cdot P1\cdot G0 + P3\cdot P2\cdot P1\cdot C0$; 2. Ri, Si — разряды i-го входа R, S ($i = 0 \ldots 3$); 3. $OVR\,(7) = (\overline{P2} + \overline{G2}\cdot\overline{P1} + \overline{G2}\cdot\overline{G1}\cdot\overline{P0} + \overline{G2}\cdot\overline{G1}\cdot\overline{G0}\cdot C0) \oplus (\overline{P3}+\overline{G3}\cdot\overline{P2}+\overline{G3}\cdot\overline{G2}\cdot\overline{P1} + \overline{G3}\cdot\overline{G2}\cdot\overline{G1}\cdot\overline{P0} + \overline{G3}\cdot\overline{G2}\cdot\overline{G1}\cdot\overline{G0}\cdot C0)$.

Блок управления формирует сигналы управления блоками МПС в соответствии с разрядами кода микрокоманды (*I8* ... *I0*). Девятиразрядную микрокоманду (МК) МПС составляют три поля: поле источника данных (*I2*, *I1*, *I0*), поле операции *АЛУ* (*I5*, *I4*, *I3*) и поле приемника *АЛУ* и сдвигов (*I8*, *I7*, *I6*).

Поле источника данных определяет возможные пары операндов *АЛУ* в соответствии с кодами, представленными в табл. 7.4. Поле операции *АЛУ* определяет операции, выполняемые МПС (см. табл. 7.5). Поле приемника *АЛУ* указывает приемник результата операции *АЛУ* и сдвиг информации в регистре *PQ* или перед записью в *РЗУ* (табл. 7.9).

При выполнении арифметических операций учитывается значение сигнала входного переноса. В табл. 7.10 приведена матрица функций и источников операндов *АЛУ*, которая показывает, что можно инкрементировать и декрементировать данные, формировать обратный и дополнительный коды, пропускать (без изменения) данные через *АЛУ* или формировать код 0 на выходе *АЛУ*. (Каждый элемент матрицы — результат на выходе *АЛУ*). Операции *АЛУ* могут быть сгруппированы на основании выполняемой функции (СЛОЖЕНИЕ, ВЫЧИТАНИЕ, И, ИЛИ и т. п.).

На рис. 7.8 показаны пути прохождения информации в МПС в зависимости от кода поля приемника микрокоманды (*I8*, *I7*, *I6*). Код поля приемника указывается в восьмеричной системе счисления. Можно выделить две группы передачи информации с выходов *АЛУ*: передача информации без сдвига (*I8* ... *I6* = $= 0 \vee 1 \vee 2 \vee 3$) и передача информации со сдвигом (*I8* ... *I6* = $= 4 \vee 5 \vee 6 \vee 7$).

При *I8* ... *I6* = 0 производится загрузка *PQ*. При *I8* ... *I6* = 1 результат операции *АЛУ* подается на выходы *Y* МПС. При *I8* ... *I6* = 2 на выходы *Y* пропускаются данные с выхода *A РЗУ*, одновременно *РЗУ* загружается значением выходов *F АЛУ*. При использовании одного из 16 РОН в качестве счетчика команд (СК), применяется код 2, чтобы загрузить регистр адреса памяти и одновременно установить в СК адрес следующей команды. При *I8* ... *I6* = 3 информация с выходов *АЛУ* загружается в *РЗУ* и передается на выходы *Y* МПС. При *I8* ... *I6* = $4 \vee 5 \vee 6 \vee 7$ результат выполнения операции в *АЛУ* передается на выходы *Y* МПС и загружается в *РЗУ*, причем при *I8* ... *I6* = 4, *I8* ... *I6* = = 6 одновременно с загрузкой в *РЗУ* осуществляется загрузка *PQ*. При коде 4 или 5 производится сдвиг в сторону младших разрядов (вправо), а при коде 6 или 7 — сдвиг в сторону старших разрядов (влево), причем при *I8* ... *I6* = 4 и *I8* ... *I6* = 6 одновременно сдвигается содержимое *АЛУ* при записи в *РЗУ* и содержимое *PQ*.

Режимы управления приемником результата операции *АЛУ* и сдвигами представлены в табл. 7.9.

Статические и динамические параметры МПС К1804ВС1 приведены в табл. 7.11 и 7.12 соответственно. На рис. 7.9 представ-

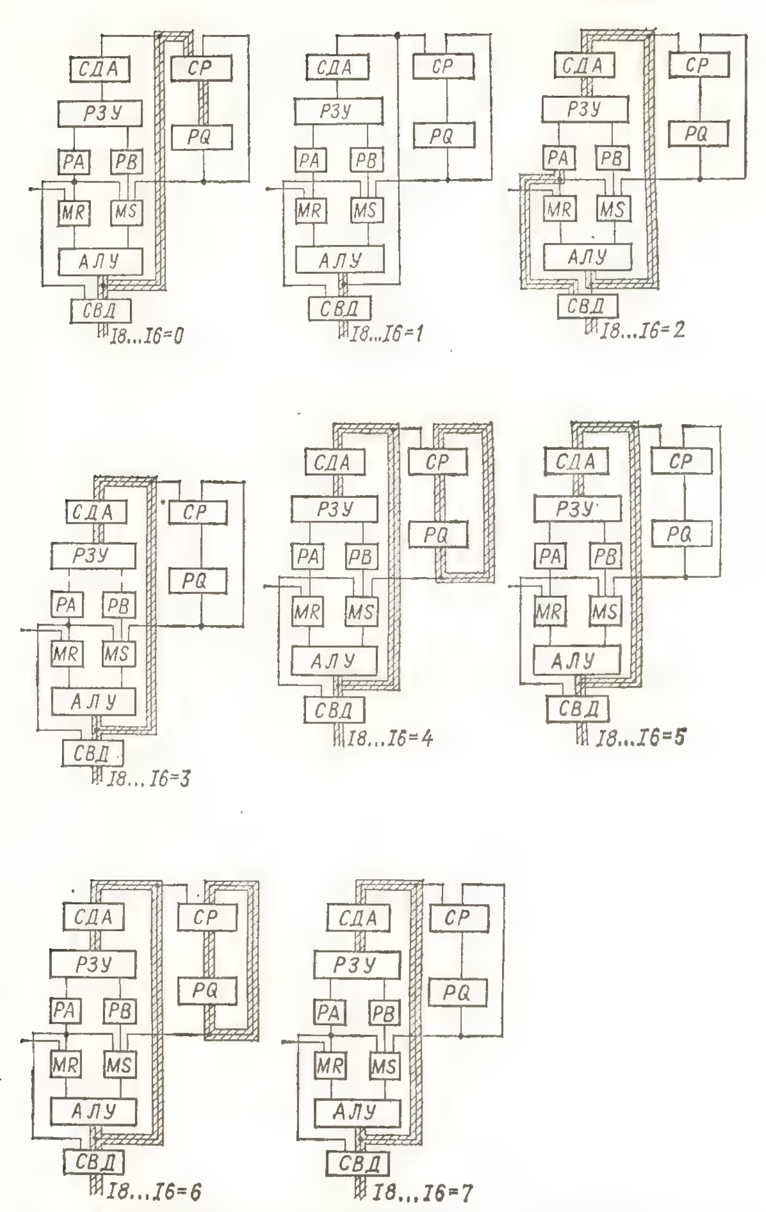


Рис. 7.8. Прохождение информации в МПС в зависимости от кода приемника информации

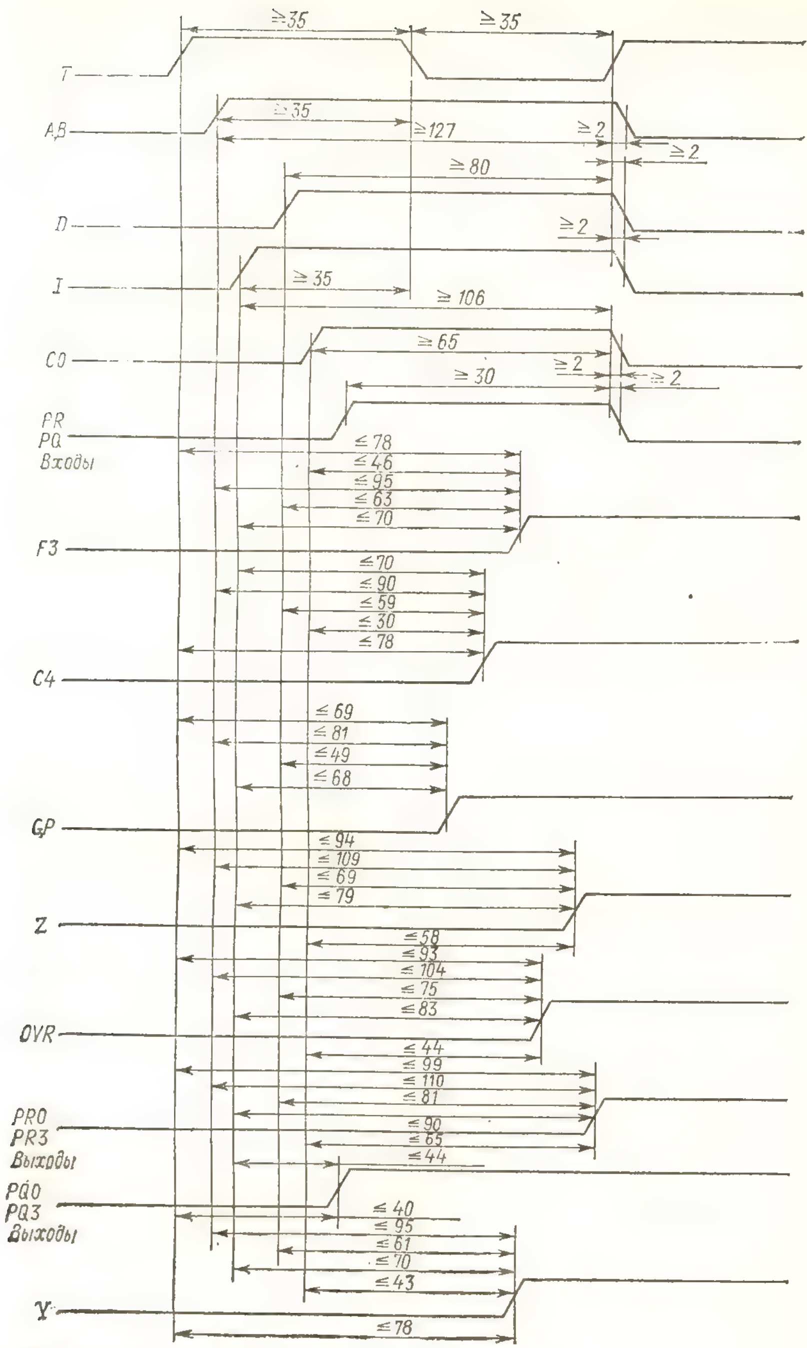


Рис. 7.9. Временна́я диаграмма работы МПС

Таблица 7.9. **Управление приемником результата АЛУ**

Микрокод				РЗУ		PQ		Выход Y	СДА		СР	
$I8$	$I7$	$I6$	8-й код	Сдвиг	Загрузка	Сдвиг	Загрузка		$PR0$	$PR3$	$PQ0$	$PQ3$
0	0	0	0	Нет	Нет	Нет	$F \to Q$	F	X	X	X	X
0	0	1	1	Нет	Нет	Нет	Нет	F	X	X	X	X
0	1	0	2	Нет	$F \to B$	Нет	Нет	A	X	X	X	X
0	1	1	3	Нет	$F \to B$	Нет	Нет	F	X	X	X	X
1	0	0	4	Вправо	$\frac{F}{2} \to B$	Вправо	$\frac{Q}{2} \to Q$	F	$F0$	Вход	$Q0$	Вход
1	0	1	5	Вправо	$\frac{F}{2} \to B$	Нет	Нет	F	$F0$	Вход	$Q0$	X
1	1	0	6	Влево	$2F \to B$	Влево	$2Q \to Q$	F	Вход	$F3$	Вход	$Q3$
1	1	1	7	Влево	$2F \to B$	Нет	Нет	F	Вход	$F3$	X	$Q3$

Примечание. Влево — в сторону старшего разряда, вправо — в сторону младшего разряда.

Таблица 7.10. **Матрица источников операндов и функций АЛУ**

*I*5 ... *I*3	*C*0	*I*2 ... *I*0			
		0	1	2	3
0	0	$A+Q$	$A+B$	Q	B
	1	$A+Q+1$	$A+B+1$	$Q+1$	$B+1$
1	0	$Q-A-1$	$B-A-1$	$Q-1$	$B-1$
	1	$Q-A$	$B-A$	Q	B
2	0	$A-Q-1$	$A-B-1$	$-Q-1$	$-B-1$
	1	$A-Q$	$A-B$	$-Q$	$-B$
3	X	$A\vee Q$	$A\vee B$	Q	B
4	X	$A\wedge Q$	$A\wedge B$	0	0
5	X	$\overline{A}\wedge Q$	$\overline{A}\wedge B$	Q	B
6	X	$A\oplus Q$	$A\oplus B$	Q	B
7	X	$\overline{A\oplus Q}$	$\overline{A\oplus B}$	$\overline{Q}$	$\overline{B}$

*I*5 ... *I*3	*C*0	*I*2 ... *I*0			
		4	5	6	7
0	0	A	$D+A$	$D+Q$	D
	1	$A+1$	$D+A+1$	$D+Q+1$	$D+1$
1	0	$A-1$	$A-D-1$	$Q-D-1$	$-D-1$
	1	A	$A-D$	$Q-D$	$-D$
2	0	$-A-1$	$D-A-1$	$D-Q-1$	$D-1$
	1	$-A$	$D-A$	$D-Q$	D
3	X	A	$D\vee A$	$D\vee Q$	D
4	X	0	$D\wedge A$	$D\wedge Q$	0
5	X	A	$D\wedge A$	$D\wedge Q$	0
6	X	A	$D\oplus A$	$D\oplus Q$	D
7	X	$\overline{A}$	$\overline{D\oplus A}$	$\overline{D\oplus Q}$	$\overline{D}$

Таблица 7.11. **Статические параметры микросхемы К1804ВС1**

Параметр, единица измерения	Обозначение	Значения параметров		Режим измерения
		мин.	макс.	
Входной ток «0», мА, для входов: *D*3 ... *D*0, *I*5 ... *I*3, *I*7 *PR*3, *PR*0, *PQ*3, *PQ*0 *C*0 *A*3 ... *A*0, *B*3 ... *B*0, *I*2 ... *I*0, *I*6, *I*8, *T*, *OE*	I_{IL}	 — — — —	 —0,72 —0,8 —3,6 —0,36	U_{CC} = 5,25 В U_{IL} = 0,5 В
Входной ток «1», мкА, для входов: *D*3 ... *D*0, *I*5 ... *I*3, *I*7 *PR*3, *PR*0, *PQ*3, *PQ*0 *C*0 *A*3 ... *A*0, *B*3 ... *B*0, *OE*, *I*2 ... *I*0, *I*6, *I*8, T	I_{IH}	 — — — —	 40 100 200 20	U_{CC} = 5,25 В U_{IH} = 2,7 В
Выходное напряжение «0», В	U_{OL}	—	0,5	I_{OL} = 16 мА (для *Z*, *G*, *Y*3 ... *Y*0) I_{OL} = 10 мА (для *C*4) I_{OL} = 8 мА (для *P*, *OVR*) I_{OL} = 6 мА (для *F*3, *PR*3, *PR*0, *PQ*3, *PQ*0)
Выходное напряжение «1», В	U_{OH}	2,4	—	I_{OH} = —1,6 мА (для *G*, *Y*3 ... *Y*0) I_{OH} = —1,0 мА (для *C*4) I_{OH} = —0,8 мА (для *P*, *OVR*) I_{OH} = —0,6 мА (для *F*3, *PR*3, *PR*0, *PQ*3, *PQ*0)
Прямое падение напряжения на антизвонном диоде, В	U_{GDI}	—	1,5	U_{CC} = 4,75 В I_I = —18 мА
Выходной ток в состоянии «выключено» при напряжении «0» на выходе, мкА, для выходов: *Y*3 ... *Y*0 *PR*3, *PR*0, *PQ*3, *PQ*0	I_{OZL}	 — —	 —50 —800	U_{CC} = 5,25 В U_{OL} = 0,4 В

Продолжение табл. 7.11

Параметр, единица измерения	Обозначение	Значения параметров		Режим измерения
		мин.	макс.	
Выходной ток в состоянии «выключено» при напряжении «1» на выходе, мкА, для выходов:	I_{OZH}			$U_{CC} = 5{,}25$ В $U_{OH} = 2{,}4$ В
$Y3 \ldots Y0$		—	50	
$PR3$, $PR0$, $PQ3$, $PQ0$		—	100	
Выходной ток «1» для выхода Z, мкА	I_{OH}	—	250	$U_{CC} = 4{,}75$ В $U_{OH} = 5{,}25$ В $U_{TH} = 2{,}0$ В $U_{IH} = 4{,}75$ В $U_{IL} = 0$ В
Ток короткого замыкания на выходе, мА	I_{OS}	—30	—85	$U_{CC} = 5{,}25$ В $U_{OL} = 0$ В
Ток потребления, мА	I_{CC}	—	280	$U_{CC} = 5{,}25$ В

Таблица 7.12. **Динамические параметры микросхемы К1804ВС1**

Параметр	Значение, нс		Температура, °С
	мин.	макс.	
Длительность тактового сигнала низкого уровня τ_L на входе T	30 35	— —	25 —10 ... 70
Длительность тактового сигнала высокого уровня τ_H на входе T	30 35	— —	25 —10 ... 70
Время задержки распространения сигнала t_p от входов A, B до выходов:			
Y	— —	85 95	25 —10 ... 70
$F3$	— —	85 95	25 —10 ... 70
$C4$	— —	80 90	25 —10 ... 70
G, P	— —	70 81	25 —10 ... 70
Z	— —	97 109	25 —10 ... 70
OVR	— —	90 104	25 —10 ... 70
$PR3$, $PR0$	— —	100 110	25 —10 ... 70

Параметр	Значение, нс		Температура, °С
	мин.	макс.	
То же от входов *D* до выходов (при арифметических операциях):			
Y		50	25
		61	−10 ... 70
*F*3	—	50	25
	—	63	−10 ... 70
*C*4	—	50	25
	—	59	−10 ... 70
G, *P*	—	40	25
	—	49	−10 ... 70
Z	—	62	25
	—	69	−10 ... 70
OVR	—	60	25
	—	75	−10 ... 70
*PR*3, *PR*0	—	70	25
	—	81	−10 ... 70
То же от входов *D* до выходов (при операции *I*8 ... *I*0 = *X*37):			
Y	—	45	25
	—	54	−10 ... 70
*F*3	—	45	25
	—	56	−10 ... 70
Z	—	57	25
	—	63	−10 ... 70
*PR*3, *PR*0	—	65	25
	—	77	−10 ... 70
То же от входа *C*0 до выходов:			
Y	—	35	25
	—	43	−10 ... 70
*F*3	—	35	25
	—	46	−10 ... 70
*C*4	—	25	25
	—	30	−10 ... 70
Z	—	52	25
	—	58	−10 ... 70
OVR	—	35	25
	—	44	−10 ... 70
*PV*3, *PR*0	—	55	25
	—	65	−10 ... 70
То же от входов *I*2 ... *I*1 до выходов:			
Y	—	60	25
	—	70	−10 ... 70
*F*3	—	60	25
	—	70	−10 ... 70
*C*4	—	55	25
	—	64	−10 ... 70
G, *P*	—	50	25
	—	59	−10 ... 70
Z	—	72	25
	—	79	−10 ... 70

Продолжение табл. 7.12

Параметр	Значение, нс		Температура, °С
	мин.	макс.	
OVR	—	70	25
	—	83	—10 ... 70
*PR*3, *PR*0	—	80	25
	—	90	—10 ... 70
То же от входов *I*5 ... *I*3 до выходов:			
Y	—	60	25
	—	70	—10 ... 70
*C*4	—	65	25
	—	78	—10 ... 70
G, *P*	—	55	25
	—	69	—10 ... 70
Z	—	82	25
	—	94	—10 ... 70
OVR	—	75	25
	—	93	—10 ... 70
*PR*3, *PR*0	—	85	25
	—	99	—10 ... 70
*PQ*3, *PQ*0	—	35	25
	—	40	—10 ... 70
Время установления сигнала t_{SU} на входах относительно положительного перепада сигнала *T* на входах:			
A, *B* (при операциях, использующих адреса *A*, *B* для чтения из *РЗУ*)	110	—	25
	127	—	—10 ... 70
D (при арифметических операциях)	75	—	25
	80	—	—10 ... 70
D (при операциях *I*8 ... *I*0=*X*37)	65	—	25
	70	—	—10 ... 70
*C*0	60	—	25
	65	—	—10 ... 70
*I*2 ... *I*1	85	—	25
	106	—	—10 ... 70
*I*5 ... *I*3	85	—	25
	100	—	—10 ... 70
*F*3	—	60	25
	—	70	—10 ... 70
*C*4	—	60	25
	—	70	—10 ... 70
G, *P*	—	55	25
	—	68	—10 ... 70
Z	—	72	25
	—	79	—10 ... 70
OVR	—	70	25
	—	81	—10 ... 70
*PR*3, *PR*0	—	80	25
	—	90	—10 ... 70
То же от входов *I*8 ... *I*6 до выходов:			
Y	—	35	25
	—	44	—10 ... 70

Параметр	Значение, нс		Температура, °C
	мин.	макс.	
*PR*3, *PR*0	—	35	25
	—	44	—10 ... 70
*PQ*3, *PQ*0	—	35	25
	—	44	—10 ... 70
То же от входов *A* до выходов *Y* (при операции *I*8 ... *I*0 = 2*XX*)	—	50	25
	—	65	—10 ... 70
То же от входа *T* до выходов:			
Y	—	65	25
	—	78	—10 ... 70
*F*3	—	65	25
	—	78	—10 ... 70
*PR*3, *PR*0, *PQ*3, *PQ*0	25	—	25
	30	—	—10 ... 70
То же относительно отрицательного перепада сигнала *T* на входах:			
A, *B* (при операциях, использующих адреса *A*, *B* для чтения из *РЗУ*)	30	—	25
	35	—	—10 ...70
B (при операциях, использующих адрес *B* для записи в РЗУ)	15	—	25
	20	—	—10 ... 70
*I*8 ... *I*6	30	—	25
	35	—	—10 ... 70
Время сохранения сигнала $t_н$ на входах *A*, *B*, *C*0, *I*2 ... *I*0, *I*5 ... *I*3, *I*8 ... *I*6, *PR*3, *RR*0, *PQ*3, *PQ*0 относительно положительного перепада сигнала *T*	0	—	25
	2	—	—10 ... 70
Время задержки распространения сигнала при переходе из состояния «выключено» в состояние «1», t_{ZH} от входа *OE* до выходов *Y*		40	25
	—	50	—10 ... 70
То же при переходе из состояния «1» в состояние «выключено» t_{HZ} от входа *OE* до выходов *Y*	—	25	25
	—	30	—10 ... 70
То же из состояния «0» в состояние «выключено» t_{LZ} от входа *OE* до выходов *Y*	—	25	25
	—	30	—10 ... 70

Примечание. Временные параметры измерялись при $U_{GG} = 5 \pm 0{,}25$ В и $C_L = 50$ пФ.

лена временная диаграмма работы МПС. МПС является тактируемым устройством, и на подачу входных сигналов накладываются определенные ограничения, строгое выполнение которых необходимо для правильного функционирования микропроцессорной секции. На диаграмме представлены времена установки сигнала перед приходом фронта переключения тактового сигнала, времена удержания сигнала после переключения тактового сигнала и времена задержки распространения сигнала от входов к выходам.

Сигналы адреса *A* и *B* считываемой информации должны быть установлены за указанное время перед приходом отрицательного фронта тактового импульса, чтобы предоставить время для доступа данных из *РЗУ* перед закрытием промежуточных регистров *PA* и *PB* в *БВП*. Адрес *A* затем может быть изменен при низком уровне тактового сигнала. Изменение адреса *B* допускается, если данные по этому адресу не записываются обратно в *РЗУ*. Если адрес *B* служит для записи данных в *РЗУ*, его изменять нельзя. Рекомендуется применять режим, при котором адреса *A* и *B* не меняются все время, пока тактовый импульс находится на уровне «0».

Время установки адресов *A* и *B* относительно положительного фронта тактового импульса необходимо для того, чтобы данные были пропущены через *БАЛ* и возвращены в *БВП*.

Сигналы *D*, *C*0, *PR*3, *PR*0, *PQ*3, *PQ*0, *I*2 ... *I*0, *I*5 ... *I*3 должны быть установлены относительно положительного фронта тактового импульса, а сигналы *I*8 ... *I*6 — относительно отрицательного фронта тактового импульса и не меняться, пока тактовый импульс находится в состоянии «0».

Наращивание. Любое число МПС может быть соединено для получения процессоров с разрядностью 8, 12, 16 и т. д. На рис. 7.10 показано соединение трех МПС, образующих 12-разрядный блок обработки данных с последовательным переносом. Шины синхронизации *T*, управления *I*8 ... *I*0 и адресов *A*3 ... *A*0, *B*3 ... *B*0 являются общими для всех секций. Сдвиговые входы-выходы *PR*3, *PQ*3 более младшей секции соединяют с входами-выходами *PR*0, *PQ*0 более старшей соседней секции. Такие соединения позволяют регистры *PQ* секций К1804ВС1 сдвигать вправо или влево как непрерывный 12-разрядный регистр, а данные на входе *РЗУ* сдвигать влево или вправо перед записью в регистры *РЗУ* как непрерывное 12-разрядное слово.

Сдвиговые входы-выходы младшей и старшей секции К1804ВС1 должны быть подключены к мультиплексору с трехстабильными выходами, который мог бы управляться микрокомандой в целях выбора подходящих сигналов для сдвиговых входов — выходов.

Выходы *Z* с открытым коллектором микросхем К1804ВС1 соединяют и через резистор R_L = 470 Ом подключают к источнику питания U_{CC} = 5,0 В. На этой линии будет «1» лишь тогда, когда на всех выходах данных из АЛУ будет «0».

Выходы переполнения *OVR* и знаковый *F*3 обычно используют в старшей секции К1804ВС1 и применяют для арифметических операций в дополнительном коде. На выходе *OVR* будет «1», когда результат арифметической операции является числом, требующим больше разрядов, чем имеется (что приведет к ошибке в знаковом разряде). Выход *F*3 — старший разряд выхода из *АЛУ*. При использовании дополнительного кода — это знак результата. В устройстве цифровой обработки данных сигналы *Z*, *OVR*, *F*3 и также выход переноса из старшей секции К1804ВС1 *C*4 используют как биты слова состояния этого устройства. Вход

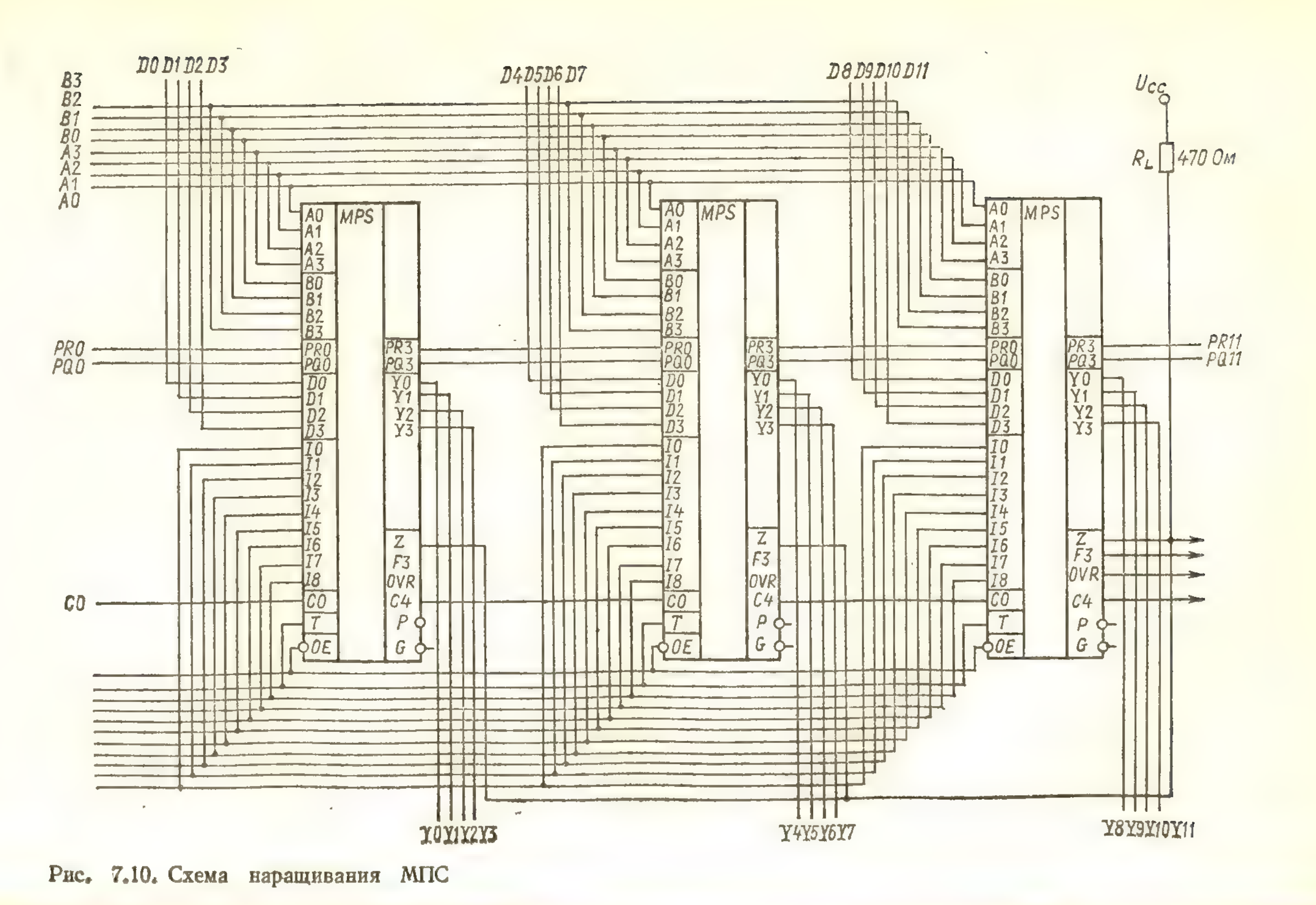


Рис. 7.10. Схема наращивания МПС

переноса младшей секции $C0$ является входом переноса для всего блока обработки данных.

Наращивание МПС можно осуществить не только с последовательным переносом (см. рис. 7.10), но и с ускоренным. При этом для такого соединения, кроме МПС, применяют схему ускоренного переноса К1804ВР1, на которую поступают сигналы $\overline{P}$ и $\overline{G}$ от МПС и сигнал $C0$. Все соединения за исключением линий $C0$ и $C4$ МПС остаются прежними. Наращивание МПС с использованием ускоренного переноса представлено ниже (см. рис. 7.19, п. 7.3).

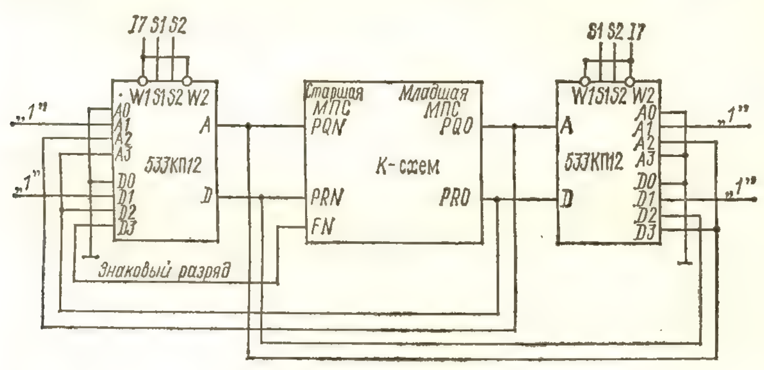


Рис. 7.11. Схема организации сдвигов

Организация сдвигов. Сдвиги результата операции АЛУ перед записью в *РЗУ* или содержимого *PQ* выполняются в соответствии со значением сигналов микрокоманды *I*8, *I*7, *I*6 (см. табл. 7.9). При соединении нескольких МПС сдвиг осуществляется с помощью двунаправленных шин *PR*3, *PR*0, *PQ*3, *PQ*0, соединенных в соответствии с правилами наращивания МПС. Однако коммутация информации на крайних МПС (младшей и старшей) должна быть организована с помощью дополнительных схем, например, мультиплексоров с трехстабильными выходами. На рис. 7.11 приведен пример организации сдвигов на блоке из *K* МПС с использованием сдвоенных четырехвходовых мультиплексоров типа 533КП12. Для управления мультиплексорами используют сигнал микрокоманды МПС *I*7, задающий направление сдвига и определяющий, какой из мультиплексоров является активным. Выходы неактивного мультиплексора находятся в выключенном состоянии и не влияют на функционирование двунаправленных шин МПС в качестве выходных шин. Сигналы управления мультиплексорами *S*1, *S*2, которые должны быть включены в состав микрокоманды, определяют тип сдвига. В табл. 7.13 описаны восемь сдвигов, реализуемых схемой на рис. 7.11. Такая схема позволяет выполнять циклические сдвиги, сдвиги слова двойной длины и арифметический сдвиг вправо. «Размножение» знака при этом обеспе-

Таблица 7.13. **Управление сдвигами**

Направление сдвига	Тип сдвига	Код			Состояние $PЗУ$ и PQ после сдвига				Схема			
					$PЗУ$ (разряд)		PQ (разряд)		$PЗУ$		PQ	
		$S1$	$S2$	$I7$	N-й	0-й	N-й	0-й	старший разряд	младший разряд	старший разряд	младший разряд
Влево (к старшему разряду)	Логический с записью «0»	0	0	1	$F(N-1)$	0	$Q(N-1)$	0				
	Логический с записью «1»	1	0	1	$F(N-1)$	1	$Q(N-1)$	1				
	Циклический	0	1	1	$F(N-1)$	FN	$Q(N-1)$	QN				
	Логический сдвиг слова двойной длины с записью «0»	1	1	1	$F(N-1)$	QN	$Q(N-1)$	0				
Вправо (к младшему разряду)	Логический с записью «0»	0	0	0	0	$F1$	0	$Q1$				
	Логический с записью «1»	1	0	0	1	$F1$	1	$Q1$				
	Циклический	0	1	0	$F0$	$F1$	$Q0$	$Q1$				
	Арифметический сдвиг слова двойной длины	1	1	0	FN	$F1$	$F0$	$Q1$				

Примечание. F — результат операции АЛУ.

чивается подачей сигнала *F*3 старшей МПС на вход *D*3 левого мультиплексора.

Вместо мультиплексоров 533КП12 можно использовать схему К1804ВР2, в которой мультиплексоры расположены в одном корпусе, предназначенную в частности для организации сдвигов на МПС К1804ВС1.

Аппаратное умножение. Для реализации умножения на МПС К1804ВС1 применяют алгоритм «сложение и сдвиг», т. е. умножение многоразрядных двоичных чисел сводится к операции сдвига и сложения. Умножение двоичных многоразрядных чисел производится путем образования частичных произведений и последующего их суммирования. Каждое частичное произведение равно нулю, если в соответствующем разряде множителя стоит нуль, равно множимому, сдвинутому на соответствующее число разрядов в сторону старших разрядов, если в разряде множителя стоит единица.

Для умножения множитель должен находиться в *PQ* микросхемы К1804ВС1, а множимое — в одном из регистров *РЗУ* по адресу *А* (РГА). Произведение расположится в другом регистре *РЗУ* по адресу *В* (РГВ). Адресные входы *А* используют для чтения множимого, а адресные входы *В* — для чтения и записи промежуточного результата (равного сумме вычисленных частичных произведений).

В каждом цикле в зависимости от значения младшего разряда *PQ* (читаемого по выходу *PQ*0) содержимое РГА складывается с содержимым РГВ либо нуль складывается с содержимым РГВ. Эту операцию выполняет *АЛУ*. В каждом цикле содержимое *PQ* и выходные данные из *АЛУ* сдвигаются в сторону младших разрядов на одну позицию. Сигналы микрокоманды для МПС на каждом цикле должны иметь следующие значения (в восьмеричном коде):

*I*8 ... *I*6 = 4 (сдвиг вправо результата *АЛУ* перед записью в *РЗУ* и *PQ*);

*I*5 ... *I*3 = 0 (сложение);

*I*2 ... *I*0 = 1 или 3 (выбор *А*, *В* или 0, *В* в качестве источников операндов для *АЛУ*).

На последнем цикле, когда проверяется старший (знаковый) разряд множителя, выполняется не сложение, а вычитание, так как знаковый разряд множителя передается не с положительным, а отрицательным арифметическим значением.

На рис. 7.12 представлена схема, реализующая умножение 8-разрядных слов. Для этого достаточно двух микропроцессорных секций. Поскольку дополнительное оборудование для умножения связано только с младшей и старшей МПС, наращивание секций для умножения *n*-разрядных слов не влияет на характер соединений схемы. Входы-выходы соседних секций соединены между собой по обычным правилам наращивания, что позволяет сдвигать *PQ* *АЛУ* сдвигать перед записью в *РЗУ* как единое слово.

Выход $PR0$ младшей секции соединен с входом $PQ3$ старшей секции. Благодаря этому младший разряд промежуточного результата на каждом цикле будет сдвигаться в старший разряд PQ. Таким образом, к концу умножения младшая половина разрядов произведения заполнит PQ, старшая — регистр *РЗУ* по адресу B.

Выход сдвига младшего разряда PQ — $PQ0$ (младший разряд множителя) определяет источники операндов для АЛУ: A и B или B и 0. Поэтому на схеме умножения инвертируемый сигнал $PQ0$ младшей секции подается на управляющий вход $I1$.

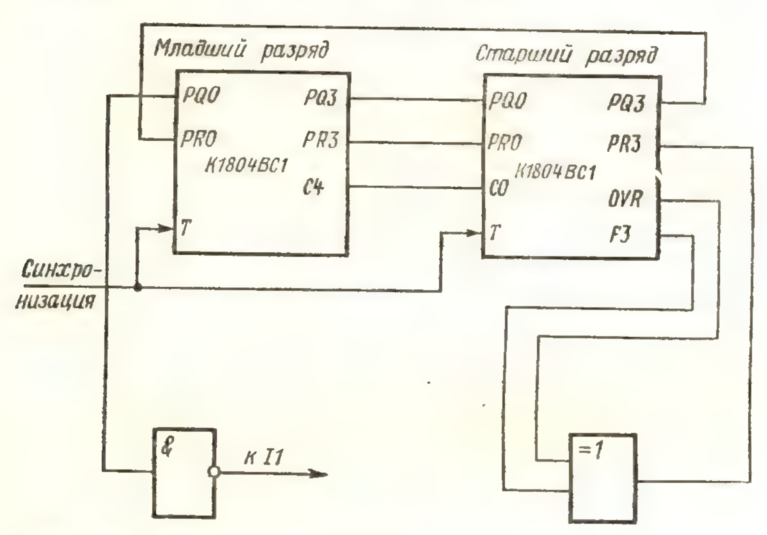


Рис. 7.12. Схема умножения 8 × 8

В то время как в каждом цикле данные на выходе *АЛУ* перед записью в *РЗУ* будут сдвигаться в сторону младших разрядов, знаковый разряд новой суммы частичных произведений будет поступать на сдвиговый вход $PR3$ старшей секции. При отсутствии сигнала переполнения (OVR = «0») на выходе $F3$ появится верный знаковый сигнал. Если во время сложения или вычитания появится сигнал переполнения (OVR = «1»), то на $F3$ появится инвертированный знаковый сигнал.

Верный знаковый сигнал для подачи на вход $PR3$ старшей секции должен быть получен операцией Исключающее ИЛИ между сигналами $F3$ и OVR. При отсутствии переполнения он будет равен $F3$, при переполнении — $\overline{F3}$. Состав и последовательность операций, выполняемых схемами МПС при умножении, приведены в табл. 7.14.

Аппаратное деление. Для реализации аппаратного деления используют алгоритм «Деление без восстановления остатка»,

Таблица 7.14. Последовательность операций МПС при умножении

Функция АЛУ	Место назначения	Описание	Число повторений	Состояние выводов (восьмеричный код)									
				A	B	$I8 \ldots I6$	$I5 \ldots I3$	$I2 \ldots I0$	$C0$	$PQ0$	$PQ3$	$PR0$	$PR3$
$0 \vee A$	PQ	Запись множителя в PQ	1	0	—	0	3	4	—	—	—	—	—
$0 \wedge A$	РЗУ	Обнуление РГ3	1	—	3	2	4	3	—	—	—	—	—
$(0+B)/2$ $(A+B)/2$	РЗУ	Условное сложение и сдвиг	n*—1	1	3	4	0	1 3	0	Выход	$PR0$	Выход	$F3 \oplus OVR$
$(B-0)/2$ $(B-A)/2$	РЗУ	Условное вычитание и сдвиг	1	1	3	4	1	1 3	1	Выход	$PR0$	Выход	$F3 \oplus OVR$
$0 \vee Q$	РЗУ	Запись разрядов произведения в РГ2	1	—	2	2	3	2	—	—	—	—	—

*n — количество разрядов множителя.

который для положительных чисел (кодов) определяет следующий порядок действий:

1) Делитель вычитают из делимого. При отрицательном результате старшая цифра частного 0, при положительном или равном нулю — 1; полученная разность в любом случае представляет собой первый остаток;

2) Предыдущий остаток удваивают;

3) Делитель прибавляют к предыдущему остатку, если остаток отрицателен, и вычитают из него, если положителен или равен нулю. При этом образуется очередной остаток, знак которого определяет очередную цифру частного по правилам, аналогичным п. 1;

4) Повторяются процедуры пп. 2, 3 до тех пор, пока не будут получены все цифры частного;

5) Прибавляют делитель, если остаток отрицательный. Истинным является остаток положительный.

Следует отметить, что если делитель имеет n разрядов, а делимое — $2n$ разрядов, то для частного требуется $(n + 1)$ разрядов. Если старший разряд делителя 0 (т. е. делитель меньшее число), то для частного потребуется больше разрядов. Для устранения этого недостатка необходимо выровнить первую 1 делителя по старшему разряду делимого путем сдвига делителя в сторону старших разрядов до тех пор, пока не исчезнут все стоящие впереди нули. В делимом они не нарушают нормального хода деления. Если перед делением было произведено такое «Выравнивание» делителя, то после деления последний остаток должен быть сдвинут в сторону младших разрядов на то же число разрядов, на какое был сдвинут делитель при выравнивании. Описанный алгоритм деления представлен на рис. 7.13.

В случае чисел со знаком необходимы некоторые изменения в алгоритме. Числа представляются в дополнительном коде, и произведенные преобразования запоминаются. После окончания деления частное и остаток также следует преобразовать в дополнительный код. Алгоритм деления для чисел со знаком представлен на рис. 7.14.

На рис. 7.15 показана схема, выполняющая деление на микропроцессорных секциях К1804ВС1. Кроме крайних секции соединяются по обычным правилам наращивания. Помимо нескольких логических элементов в схеме аппаратного деления в качестве регистра состояния применен регистр К1804ИР1 (см. 7.6). Следует только учесть, что здесь используется один разряд этого регистра, имеющего обычный выход Q и выход Y с тремя устойчивыми состояниями, который управляется сигналом $\overline{OE}$. Когда выход Y разрешен, то на выходах Q и Y — идентичные сигналы.

Пусть делимое находится в регистре РГ0 *РЗУ* (оно будет потеряно при делении, и его место займет остаток), делитель — в регистре РГ1. *PQ* должен быть очищен, т. е. подготовлен для записи в него частного.

После проверки знаков делимого и делителя ($I5 \dots I0 = 23_8$ и 24_8) делитель выравнивается ($I5 \dots I0 = 33_8$). Старший его разряд записывается в регистр состояния. Сдвиги при выравнивании осуществляются с помощью $I8 = I6 =$ «1» и $I7$ равного эквивалентности текущего и предыдущего старшего разряда делителя (см. рис. 7.15). Если они оба равны «0», то $I7 =$ «1» и будет осуществлен сдвиг в сторону старших разрядов. Если $I7 =$ «0» (контролируемые разряды различны), то сдвиг осуществляется в стороны младших разрядов.

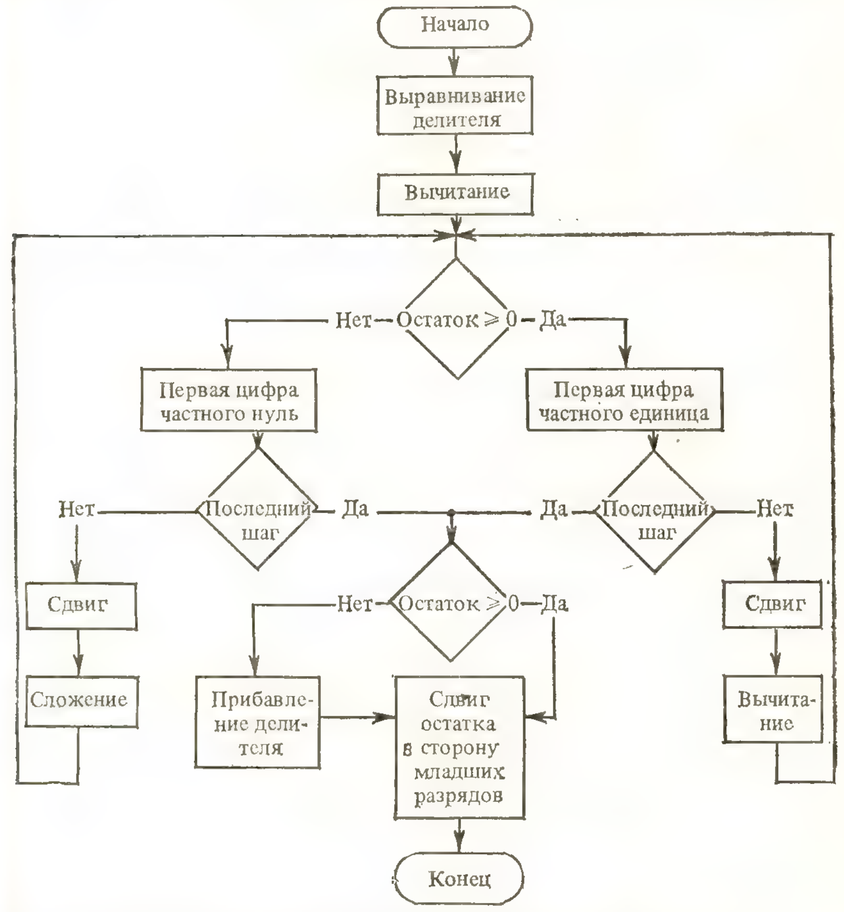


Рис. 7.13. Алгоритм деления без восстановления остатка для кодов

В то же время с помощью выхода Y регистра состояния может быть восстановлен знаковый разряд (по входу $PR3$).

Первым шагом деления является вычитание. Затем делимое в РГ0 и PQ сдвигают в сторону старших разрядов. При этом

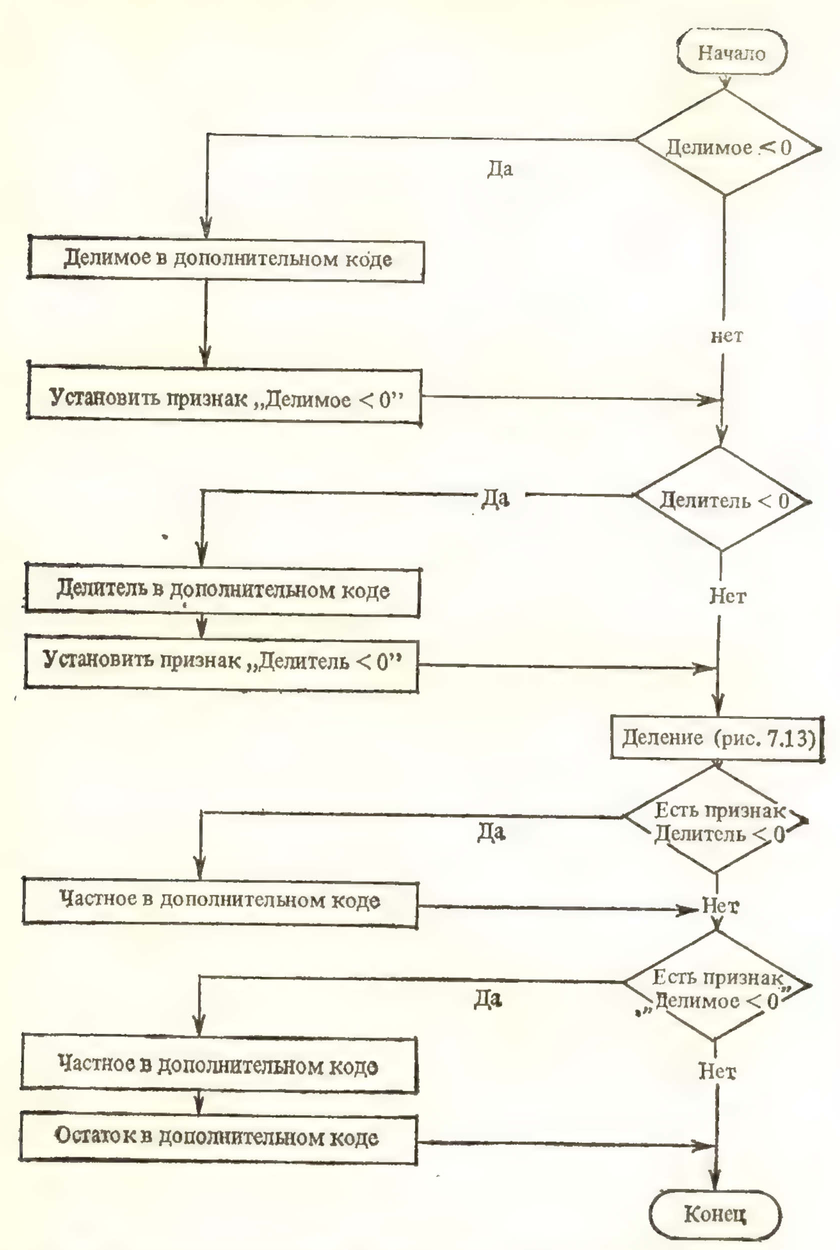


Рис. 7.14. Алгоритм деления без восстановления остатка для чисел

$I8 \ldots I6 = 6_8$, $I2 \ldots I0 = 1_8$, $I5 \ldots I4 = 0_8$. Если разряд CL микрокоманды «1», то оба сигнала $I3 = C0 = 1$ и *АЛУ* выполняет вычитание над числами в дополнительном коде. Знак остатка будет записан в регистр состояния, а его инверсное значение будет записано в младший разряд PQ при его сдвиге в сторону старших разрядов.

Эта процедура повторяется для всех других разрядов остатка. Разряд CL микрокоманды, будучи равен «0», разрешает предыдущему знаковому биту определять сигнал $I3$. Если он равен «0»

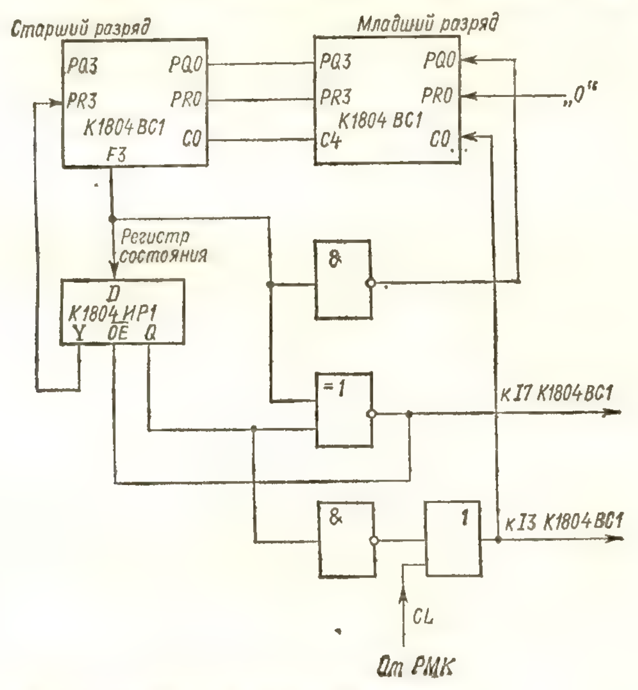


Рис. 7.15. Схема аппаратного деления

(т. е. остаток больше или равен нулю), то $I3$ и $C0$ равны «1» и *АЛУ* выполняет вычитание; если он равен «1» (остаток меньше нуля), то $I3$ и $C0$ равны «0» и *АЛУ* выполняет сложение. При каждом сдвиге в сторону старших разрядов инверсное значение знакового разряда будет записываться в младший разряд частного.

При завершении деления необходимо рассмотреть знак остатка, и если в знаковом разряде «1», к остатку необходимо прибавить делитель. Для этого выполняют безусловное сложение ($C0$ = «0»). Причем $I2$ = «0», $I0$ = «1», а $I1$ управляется инверсным значением знака остатка, т. е. выполняется прибавление к РГ0 либо РГ1, если знаковый разряд остатка «1», либо нуля, если знаковый разряд остатка есть «0».

Таблица 7.15. Операции МПС для реализации деления

Функция АЛУ	Место назначения	Описание	CL	Число повторений	Состояние выводов A	B	$I8 \ldots I6$	$I5 \ldots I3$	$I2 \ldots I0$	$C0$	$PQ0$	$PQ3$	$PR0$	$PR3$
$(0 \vee B)/2$	РЗУ	Выравнивание делителя	1	K *	—	1	5	3	3	—	—	—	0	Знак
$(0 \vee B) \cdot 2$							7							—
$(B - A) \cdot 2$	РЗУ	Первое вычитание и сдвиг	1	—	1	0	6	1	1	1	$\overline{F3}$	—	0	—
$(B - A) \cdot 2$	РЗУ	Вычитание, сложение и сдвиг	0	K	1	0	6	1	1	1	$\overline{F3}$	—	0	—
$(B + A) \cdot 2$								0		0				
$B + A$	РЗУ	Поправка остатка	—	—	1	0	3	0	1	0	—	—	—	—
$B + 0$									3					
$B + 0$	РЗУ	Сдвиг остатка	1	K	—	1	5	1	3	1	—	—	—	Знак

* K — число впередистоящих нулей делителя.

Таблица 7.16. **Содержимое регистров при делении**

Регистр	Исходное	Конечное
РГ0 РГ1 PQ	Делимое Делитель Очищен	Остаток Делитель Частное

В заключительной фазе деления в случае необходимости производится выравнивание остатка и преобразование частного и остатка в дополнительный код в зависимости от признаков знака делимого и делителя.

Последовательность операций, выполняемых МПС для реализации деления, представлена в табл. 7.15. В табл. 7.16 отображено исходное и конечное содержимое регистров МПС, используемых при делении.

Перестановка байтов или полуслов. Чтобы поменять местами левую и правую части слова или старший и младший байты в случае 16-разрядного слова с помощью *РЗУ* производят сдвиг слова на два разряда в одном цикле. При этом для перестановки байтов 16-разрядного слова потребуется всего четыре цикла.

Для выполнения перестановки соединения МПС должны быть организованы как для циклического сдвига. Вывод *C*4 старшей МПС должен быть соединен с *C*0 младшей. Содержимое одного и того же регистра *РЗУ* вызывается по адресам *A* и *B*, *АЛУ* выполняет операцию сложения, и перед записью в *РЗУ* результат сдвигается в сторону старших разрядов.

7.3. СХЕМА УСКОРЕННОГО ПЕРЕНОСА К1804ВР1

Схема ускоренного переноса (СУП) предназначена для организации параллельного переноса в операционных блоках, построенных на микропроцессорных секциях К1804ВС1 или других схемах, имеющих аналогичные выходы предварительного просмотра переноса. Одна СУП позволяет организовать параллельные цепи переноса в 16-разрядном операционном блоке из четырех МПС К1804ВС1. Каскадное включение схем К1804ВР1 позволяет организовать параллельные цепи переноса в операционных блоках с разрядностью более 16.

Условное графическое обозначение СУП приведено на рис. 7.16. Назначение выводов описано в табл. 7.17.

Функциональная схема СУП К1804ВР1 приведена на рис. 7.17. СУП является комбинационной схемой, и формирование сигналов на выходах *CX*, *CY*, *CZ*, *P* и *G* происходит согласно следующим выражениям:

$$\overline{CX} = \overline{C0} \cdot G0 + P0 \cdot G0;$$

$$\overline{CY} = \overline{C0} \cdot G0 \cdot G1 + P0 \cdot G0 \cdot G1 + P1 \cdot G1;$$

$$\overline{CZ} = \overline{C0} \cdot G0 \cdot G1 \cdot G2 + P0 \cdot G0 \cdot G1 \cdot G2 + P1 \cdot G1 \cdot G2 + P2 \cdot G2;$$

$$P = P0 + P1 + P2 + P3;$$

$$G = P3 \cdot G3 + P2 \cdot G2 \cdot G3 + P1 \cdot G1 \cdot G2 \cdot G3 +$$
$$+ G0 \cdot G1 \cdot G2 \cdot G3.$$

Входные сигналы генерации переноса ($G3 \ldots G0$) и распространения переноса ($P3 \ldots P0$) поступают на СУП с четырех последо-

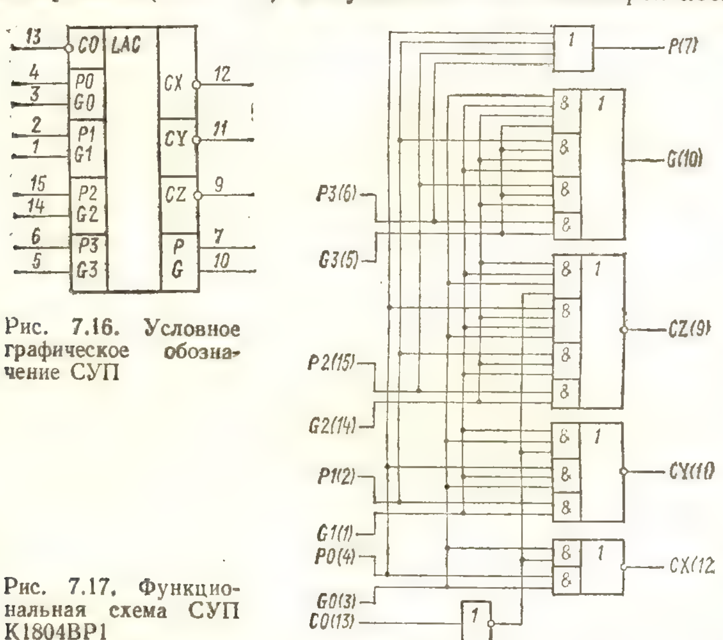


Рис. 7.16. Условное графическое обозначение СУП

Рис. 7.17. Функциональная схема СУП К1804ВР1

вательно соединенных групп элементов, выполняющих функции АЛУ. Выходные сигналы G и P предназначены для каскадирования и подаются на входы $G3 \ldots G0$ и $P3 \ldots P0$ схем К1804ВР1 следующего уровня организации параллельного переноса. Входной сигнал $C0$ предназначен наряду с сигналами генерации и распространения переноса для формирования сигналов переноса на выходах CX, CY, CZ. Сигнал $C0$ должен подаваться на СУП одновременно с подачей его на МПС К1804ВС1 или другие схемы типа АЛУ. Сигналы выходного переноса СУП, CX, CY, CZ являются входными сигналами переноса для последующих МПС (или групп АЛУ) и подаются на входы $C0$ МПС. Сигналы CX, CY, CZ формируются независимо друг от друга, и на значение каждого из них влияют только сигналы на входе переноса ($C0$) и соответствующих входах распространения и генерации переноса.

Статические и динамические параметры СУП приведены в табл. 7.18 и табл. 7.19 соответственно. Временная диаграмма работы СУП представлена на рис. 7.18.

Использование СУП позволяет существенно сократить время суммирования многоразрядных слов за счет организации параллельных цепей переноса по сравнению с последовательным переносом. На рис. 7.19 приведен пример построения 16-разрядного операционного блока с ускоренным переносом на четырех МПС К1804ВС1 и одной СУП К1804ВР1. При длине слова более 16 разрядов существует два варианта использования СУП. При первом варианте СУП применяют для организации параллельного переноса внутри 16-разрядных групп секций и последовательного между группами. При втором — осуществляют каскадирование, обеспечивающее многоуровневую организацию

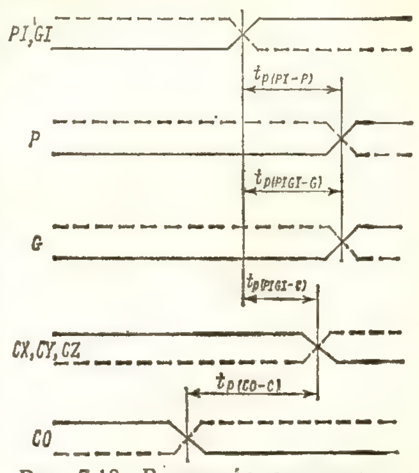


Рис. 7.18. Временна́я диаграмма работы СУП

Таблица 7.17. **Назначение выводов микросхемы К1804ВР1**

Номер вывода	Назначение	Обозначение	Разряд	Тип
1	Сигнал генерации переноса	*G*1	1	Вход
2	Сигнал распространения переноса	*P*1	1	Вход
3	Сигнал генерации переноса	*G*0	0	Вход
4	Сигнал распространения переноса	*P*0	0	Вход
5	Сигнал генерации переноса	*G*3	3	Вход
6	Сигнал распространения переноса	*P*3	3	Вход
7	Сигнал распространения переноса	*P*	—	Выход
8	Общий	*GND*	—	—
9	Сигнал переноса старшей группы	*CZ*	—	Выход
10	Сигнал генерации переноса	*G*	—	Выход
11	Сигнал переноса средней группы	*CY*	—	Выход
12	Сигнал переноса младшей группы	*CX*	—	Выход
13	Перенос	*C*0	—	Вход
14	Сигнал генерации переноса	*G*2	2	Вход
15	Сигнал распространения переноса	*P*2	2	Вход
16	Питание	U_{cc}	—	—

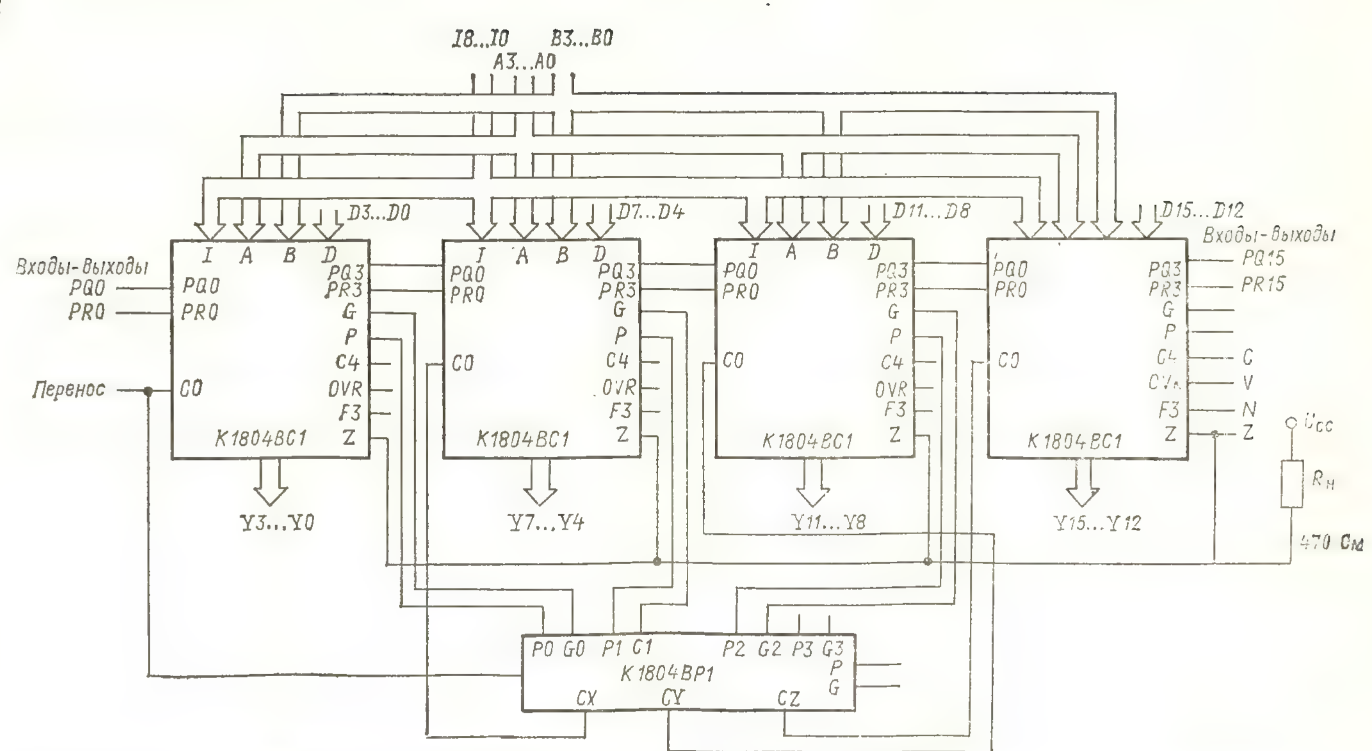


Рис. 7.19. Схема соединения четырех МПС К1804ВС1 с применением СУП К1804ВР1

Таблица 7.18. **Статические параметры микросхемы К1804ВР1**

Параметр, единица измерения	Обозначение	Значения параметров		Режим измерения
		мин.	макс.	
Входной ток «0», мА для входов:	I_{IL}			$U_{CC} = 5{,}25$ В $U_{IL} = 0{,}5$ В
$C0$		—	—2	
$P3$		—	—4	
$P2$		—	—6	
$P1$, $P0$, $G3$		—	—8	
$G0$, $G2$		—	—14	
$G1$		—	—16	
Входной ток «1», мкА для входов:	I_{IH}			$U_{CC} = 5{,}25$ В $U_{IH} = 2{,}7$ В
$C0$		—	50	
$P3$, $P2$, $P1$, $P0$, $G3$		—	200	
$G2$, $G1$, $G0$		—	400	
Выходное напряжение «0», В	U_{OL}	—	0,5	$U_{CC} = 4{,}75$ В $U_{TL} = 0{,}8$ В $U_{IL} = 0$ В $U_{IH} = 4{,}75$ В $I_{OL} = 20$ мА
Выходное напряжение «1», В	U_{OH}	2,4	—	$U_{CC} = 4{,}75$ В $U_{TH} = 2{,}0$ В $U_{IL} = 0$ В $U_{IH} = 4{,}75$ В $I_{OH} = -1$ мА
Прямое падение напряжения на антизвонном диоде, В	U_{GDI}	—	1,2	$U_{CC} = 4{,}75$ В $I_I = -18$ мА
Ток короткого замыкания на выходе, мА	I_{OS}	—40	—100	$U_{CC} = 5{,}25$ В $U_{OL} = 0$ В
Ток потребления, мА	I_{CC}	—	109	$U_{CC} = 5{,}25$ В

Таблица 7.19. **Динамические параметры микросхемы К1804ВР1**

Параметр	Максимальное значение, нс	Температура, °С
Время задержки распространения сигнала		
от входа $C0$ до выходов CX, CY, CZ	19	25
	20	—10 ... 70
от входов $G2 \dots G0$, $P2 \dots P0$ до выходов CX, CY, CZ	15	25
	16	—10 ... 70
от входов $G3 \dots G0$, $P3 \dots P1$ до выхода G	19	25
	20	—10 ... 70
от входов $P3 \dots P0$ до выхода P	19	25
	20	—10 ... 70

Примечание. Временные параметры измерялись при $U_{CC} = 5 \pm 0{,}25$ В и $C_L = 50$ пФ.

цепей переноса, т. е. параллельный перенос на всех уровнях, как внутри 16-разрядных групп, так и между группами любой разрядности. Это дает возможность получить операционный блок, имеющий наибольшее быстродействие. На рис. 7.20 представлен 32-разрядный операционный блок с каскадным включением СУП, имеющий 8 схем К1804ВС1 и 3 схемы К1804ВР1.

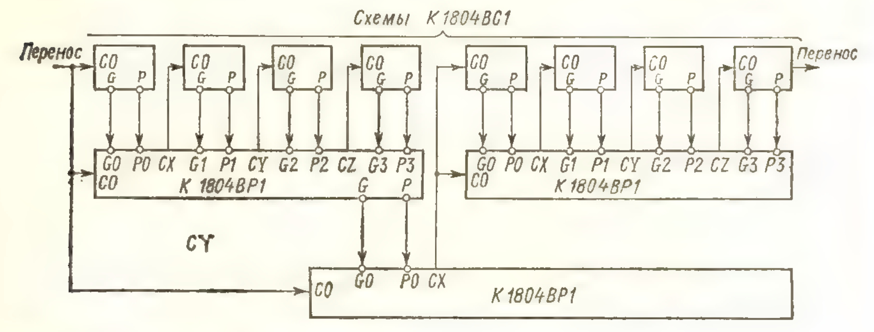


Рис. 7.20. Каскадное соединение СУП

7.4. СЕКЦИИ УПРАВЛЕНИЯ АДРЕСОМ МИКРОКОМАНДЫ К1804ВУ1 И К1804ВУ2

Секции управления адресом микрокоманды (СУАМ) предназначены для управления последовательностью выборки микрокоманд из памяти микропрограмм и формирования адреса микрокоманды под воздействием внешних и внутренних сигналов. Назначение микросхем К1804ВУ1 (СУАМ1) и К1804ВУ2 (СУАМ2) одинаково, поэтому обе схемы рассматриваются в одном разделе. Различия между ними будут показаны по ходу описания их структуры и функционирования. СУАМ имеют следующие архитектурные особенности: разрядность 4 и возможность наращивания до любой разрядности, кратной четырем; внутренний регистр адреса; стек глубиной 4, обеспечивающий переход с возвратом при выполнении микропрограмм; наращиваемый 4-разрядный счетчик микрокоманд; наличие входа установки нулевого адреса для начальной выборки микрокоманды по этому адресу; наличие входов маски для модификации адреса (только для СУАМ1); наличие трехстабильных выходов.

Условные графические обозначения СУАМ1 (К1804ВУ1) и СУАМ2 (К1804ВУ2) представлены на рис. 7.21 и рис. 7.22 соответственно. Назначение выводов СУАМ1 дано в табл. 7.20, а СУАМ2 в табл. 7.21.

Структурные схемы СУАМ1 и СУАМ2 приведены на рис. 7.23 и 7.24. В структурных схемах обеих СУАМ можно выделить следующие узлы: регистр адреса *РА*, блок выборки адреса *БВА*, счетчик микрокоманд *СМК*, *Стек*, буфер адреса *БА*. Рассмотрим подробно их устройство и функционирование.

Регистр *РА* предназначен для хранения адреса, принятого от внешнего источника по входам *R*3 ... *R*0 (для СУАМ1) или входам *D*3 ... *D*0 (для СУАМ2). *РА* состоит из четырех триггеров *D*-типа и управляется сигналами *T* и $\overline{RE}$. При подаче «0» на вход разрешения записи в регистр адреса (*RE*) по положительному фронту тактового сигнала *T* происходит запись информации. При $\overline{RE}$ = = «1» запись информации с внешних шин не производится, а по каждому положительному фронту сигнала *T* подтверждается хранимая в регистре информация.

Блок *БВА* выбирает источник адреса следующей микроко-

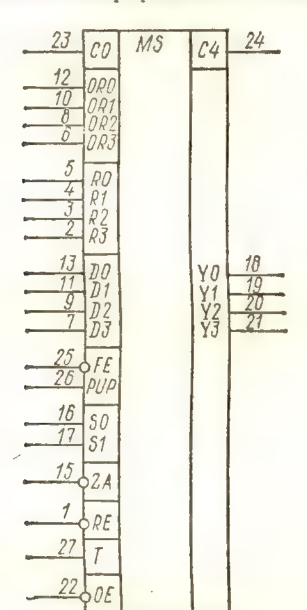


Рис. 7.21. Условное графическое обозначение К1804ВУ1

Рис. 7.22. Условное графическое обозначение К1804ВУ2

манды. В зависимости от значений управляющих сигналов *S*1 и *S*0 мультиплексора, входящего в состав *БВА*, в качестве адреса следующей микрокоманды может быть адрес с прямых входов *D*3 ... *D*0, содержимое регистра адреса, накопителя стека либо содержимое счетчика микрокоманд. В *БВА* СУАМ1 четыре входа маски *OR*3 ... *OR*0 поразрядно соединены с выходами мультиплексора *БВА* по схеме ИЛИ.

Единичное значение сигнала на входе маски устанавливает в «1» выход соответствующего разряда СУАМ (при наличии «1» на входе *ZA*). Сигнал со входа *ZA* объединен по И с каждым информационным разрядом *БВА* (как в СУАМ1, так и в СУАМ2). Подача «0» на вход *ZA* приводит к установке «0» на выходах *Y*. Вход *ZA* используется для повторного запуска микропрограмм с нулевого адреса.

Состояния входов и выхода Y одного из разрядов СУАМ, поясняющие работу *БВА*, приведены в табл. 7.22. Значения выходов Y даны при условии разрешения выдачи информации через *БА*.

Счетчик микрокоманд (см. рис. 7.23) включает *Регистр СМК* и схему прибавления единицы (*Инкрементор*). *Регистр СМК* представляет собой четырехразрядный регистр, состоящий из триггеров D-типа. Запись информации в регистр происходит по положительному фронту сигнала T. Информация с выхода регистра *СМК* передается в *Стек* и в *БВА* (вход M).

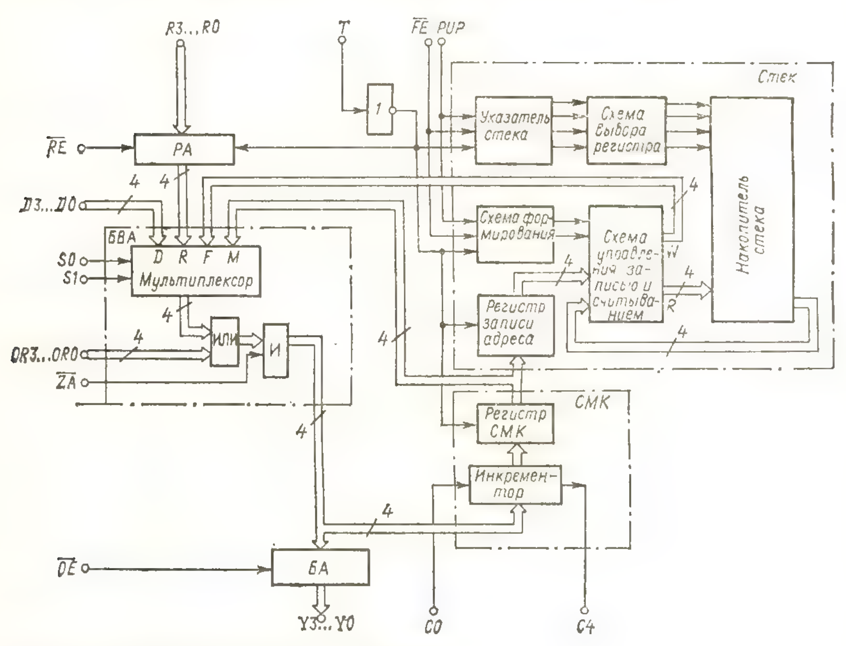


Рис. 7.23. Структурная схема К1804ВУ1

Инкрементор имеет вход переноса $C0$. При «1» на входе $C0$ в регистр СМК записывается значение адреса, поступающее с *БВА* на *СМК*, увеличенное на единицу. Благодаря этому в *СМК* формируется адрес следующей микрокоманды. При «0» на входе $C0$ значение адреса, поступающее с *БВА* на *СМК*, не модифицируется. Одна и та же команда может быть выполнена любое число раз.

Выход $C4$ инкрементора предназначен для непосредственного наращивания СУАМ для получения большей разрядности. При «1» на входе $C0$ и на всех остальных входах *СМК Инкрементор* формирует единичное значение сигнала на выходе переноса $C4$ счетчика микрокоманд.

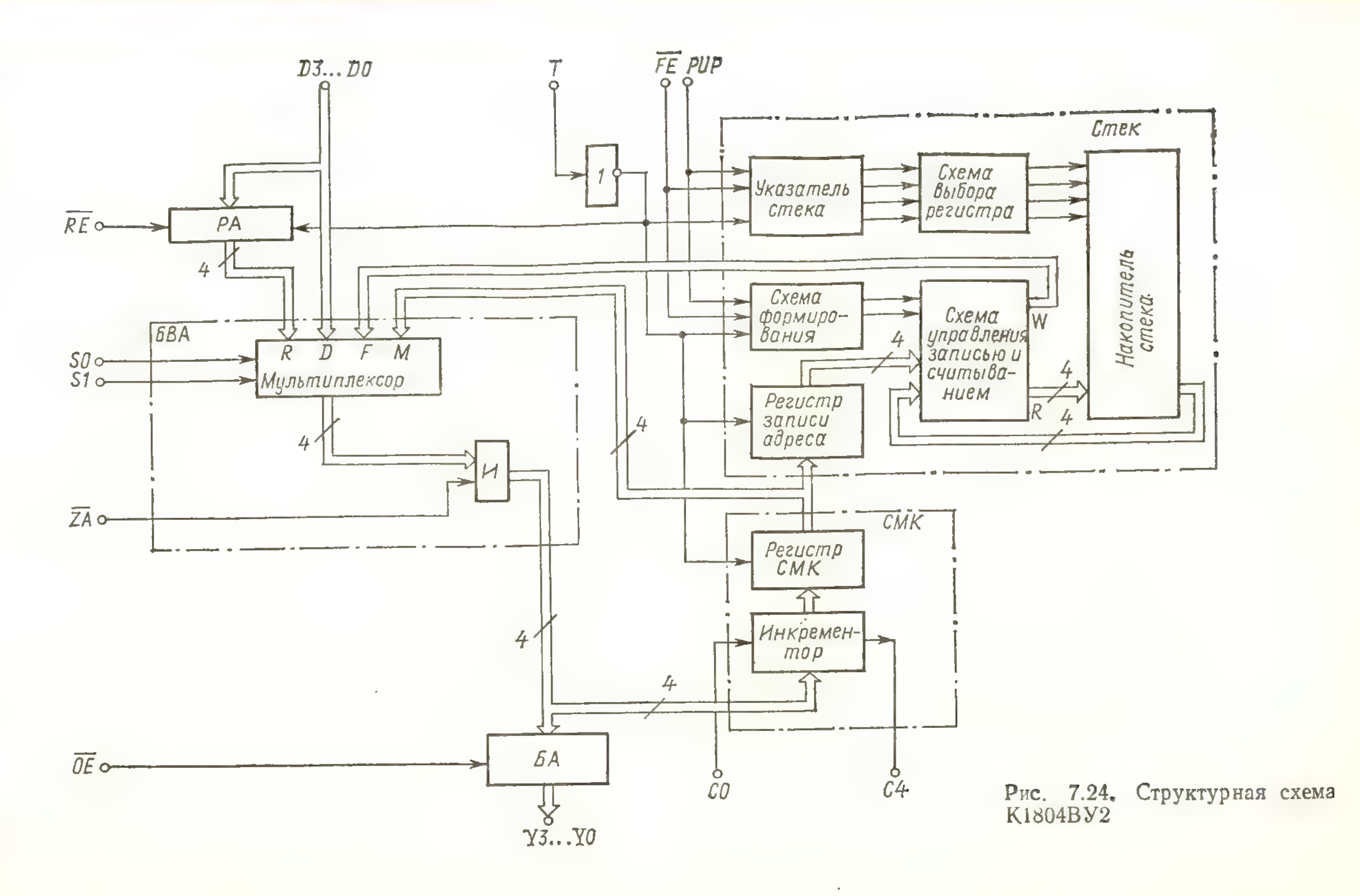


Рис. 7.24. Структурная схема К1804ВУ2

Таблица 7.20. Назначение выводов микросхемы К1804ВУ1

Номер вывода	Назначение	Обозначение	Разряд	Тип
1	Разрешение записи в регистр адреса	*RE*	—	Вход
2	Данные регистра адреса	*R*3	3	
3		*R*2	2	
4		*R*1	1	
5		*R*0	0	
6	Маска	*OR*3	3	
7	Адрес	*D*3	3	
8	Маска	*OR*2	2	
9	Адрес	*D*2	2	
10	Маска	*OR*1	1	
11	Адрес	*D*1	1	
12	Маска	*OR*0	0	
13	Адрес	*D*0	0	
14	Общий	*GND*	—	—
15	Установка нулевого адреса	*ZA*	—	Вход
16	Выбор адреса	*S*0	0	
17		*S*1	1	
18	Адрес	*Y*0	0	Выход
19		*Y*1	1	
20		*Y*2	2	
21		*Y*3	3	
22	Разрешение выбора адреса	*OE*	—	Вход
23	Перенос в счетчик микрокоманд	*C*0	—	
24	Перенос из счетчика микрокоманд	*C*4	—	Выход
25	Разрешение управления стеком	*FE*	—	Вход
26	Управление стеком	*PUP*	—	
27	Тактовый	*T*	—	
28	Питание	U_{CC}	—	—

Примечание. Выходы *Y*3 ... *Y*0 трехстабильные

Стек СУАМ (см. рис. 7.23, 7.24) состоит из указателя стека, накопителя стека, схемы выбора регистра накопителя стека, схемы формирования сигнала разрешения записи в стек, схемы управления записью и считыванием и регистра записи адреса. Фрагмент функциональной схемы СУАМ, представляющей стек, приведен на рис. 7.25.

Указатель стека предназначен для выбора регистра накопителя стека, к которому производится обращение. Он представляет собой двухразрядный двоичный реверсивный счетчик, построенный на триггерах *D*-типа. Запись информации в них происходит по положительному фронту тактового импульса *T*.

Управление стеком осуществляется сигналами на входе *FE* и *PUP*. При подаче «0» на вход *FE* разрешается изменение указателя стека. При этом, если на входе *PUP* имеем «0», то по положительному фронту тактового импульса *T* происходит уменьше-

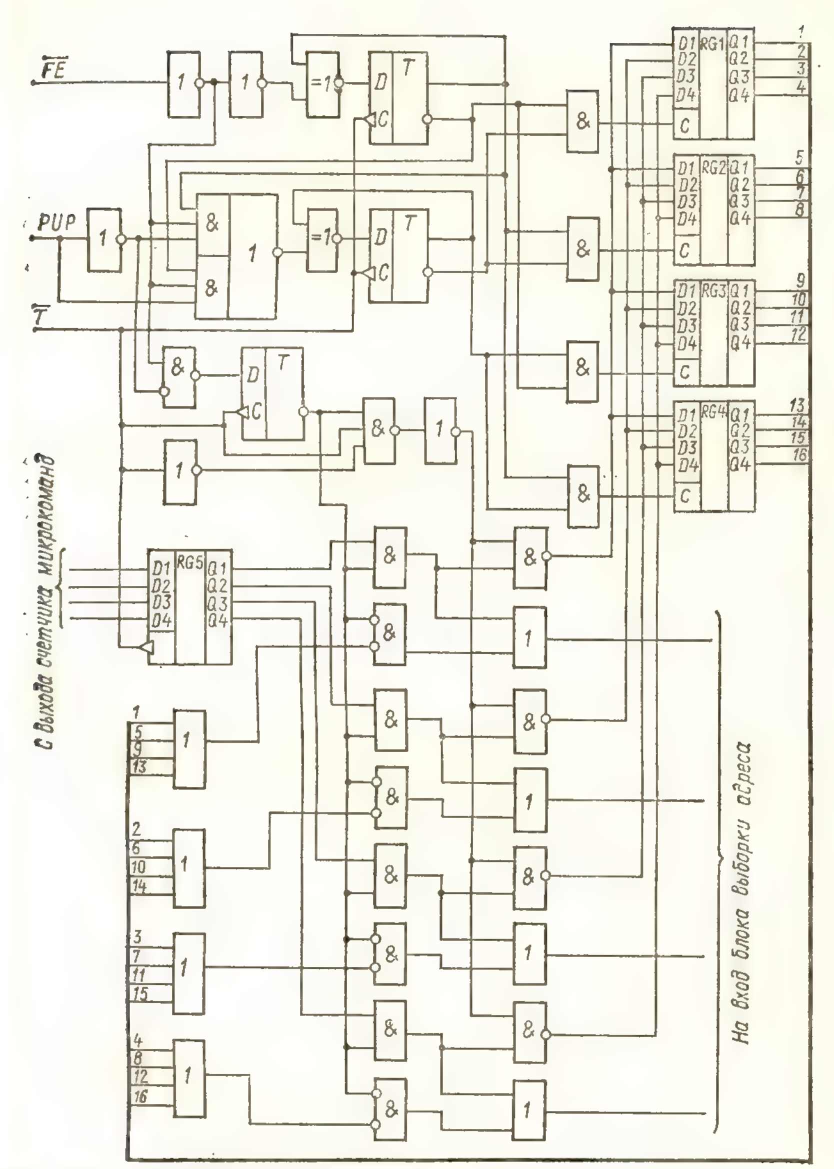


Рис. 7.25. Функциональная схема стека

Таблица 7.21. **Назначение выводов микросхемы К1804ВУ2**

Номер вывода	Назначение	Обозначение	Разряд	Тип
1	Тактовый	T	—	Вход
2	Питание	U_{CC}	—	—
3	Разрешение записи в регистр адреса	RE	—	Вход
4	Данные регистра адреса	$D3$	3	Вход
5	Данные регистра адреса	$D2$	2	Вход
6	Данные регистра адреса	$D1$	1	Вход
7	Данные регистра адреса	$D0$	0	Вход
8	Общий	GND	—	—
9	Установка нулевого адреса	ZA	—	Вход
10	Выбор адреса	$S0$	0	Вход
11	Выбор адреса	$S1$	1	Вход
12	Адрес	$Y0$	0	Выход
13	Адрес	$Y1$	1	Выход
14	Адрес	$Y2$	2	Выход
15	Адрес	$Y3$	3	Выход
16	Разрешение выбора адреса	OE	—	Вход
17	Перенос в счетчик микрокоманд	$C0$	—	Вход
18	Перенос из счетчика микрокоманд	$C4$	—	Выход
19	Разрешение управления стеком	FE	—	Вход
20	Управление стеком	PUP	—	Вход

Примечание. Выходы $Y3 \ldots Y0$ трехстабильные.

ние указателя стека следующим образом: $2 \to 1 \to 0 \to 3$ (с возвратом от 3 к 2). Если же на входе FE «0» и на входе PUP «1», то по положительному фронту тактового импульса T происходит увеличение указателя стека следующим образом: $2 \to 3 \to 0 \to 1$ (с возвратом от 1 к 2).

Таблица 7.22. **Управление мультиплексором выхода СУАМ**

$S1$	$S0$	Вход мультиплексора БВА	ORi	ZA	Y_i
X	X	X	X	0	0
X	X	X	1	1	1
0	0	M	0	1	$[СМК]_i$
0	1	R	0	1	$[РА]_i$
1	0	F	0	1	$[СТ0]_i$
1	1	D	0	1	$[D]_i$

Примечание. Здесь приняты следующие обозначения: X — безразличное состояние сигнала; $[СМК]_i$ — содержимое i-го разряда счетчика микрокоманд; $[РА]_i$ — содержимое i-го разряда регистра адреса; $[СТ0]_i$ — содержимое i-го разряда накопителя стека, адресуемого указателем стека; $[D]_i$ — значение i-го разряда прямого входа.

Таблица 7.23. Управление стеком

Управляющие сигналы		Режим работы стека	
$\overline{FE}$	PUP	Указатель	Накопитель
1	Безразлично	Не изменяется	Считывание
0	1	Увеличение указателя	Запись
0	0	Уменьшение указателя	Считывание

Указатель стека всегда указывает регистр накопителя стека, в котором находится последнее записанное в стек слово.

Схема выбора регистра накопителя стека представляет дешифратор сигналов, поступающих с указателя стека. С выхода этого дешифратора сигналы поступают на тактовые входы регистров накопителя стека (см. рис. 7.25).

Накопитель стека — это четыре 4-разрядных регистра на триггерах *D*-типа со статической записью. Запись выполняется в течение интервала времени, когда тактовый сигнал *T* имеет низкий уровень.

Схема формирования сигнала разрешения записи вырабатывает сигнал разрешения записи в *Накопитель стека* по отрицательному фронту импульса Т при «0» на входе FE и «1» на входе PUP.

Регистр записи адреса *RG*5 (см. рис. 7.25) состоит из четырех триггеров *D*-типа. Запись информации в них происходит по положительному фронту сигнала *T*. Этот регистр служит для предварительного хранения адреса, передаваемого из *СМК* в накопитель стека.

Схема управления записью и считыванием обеспечивает необходимую временную и логическую коммутацию внутренних цепей стека под воздействием сигналов *T*, $\overline{FE}$, *PUP* для выполнения передачи информации из *СМК* через регистр записи адреса в накопитель стека (через выход *R* схемы управления) или из накопителя стека на вход *БВА* (через выход *W* схемы управления) при считывании (см. рис. 7.23, 7.24).

Режимы работы стека под воздействием управляющих сигналов приведены в табл. 7.23. На рис. 7.26 представлена временная диаграмма управления стеком.

Буфер адреса состоит из четырех логических элементов, имеющих выходы с тремя состояниями. При подаче «0» на управляющий вход *OE* адрес с выхода *БВА* передается на выходы СУАМ *У*3 ... *У*0. При подаче «1» на вход *OE* выходы СУАМ отключаются (находятся в состоянии высокого сопротивления).

Рассмотрим последовательность выработки адресов в СУАМ в зависимости от значений управляющих сигналов $S1$, $S0$, $C0$, $OR3 \ldots OR0$, PUP и $\overline{FE}$. При подаче «1» на вход $C0$ происходит последовательное увеличение адреса на единицу, а при подаче «1» на i-й вход маски ($OR3 \ldots OR0$) адрес может быть изменен на величину 2^i (i — вес разряда маски). Сигналы на входах $S1$, $S0$ определяют, какой из четырех источников адреса выбрать в качестве следующего адреса микрокоманды. А сигналы PUP и $\overline{FE}$ определяют три режима работы стека. Комбинируя сигналами $S1$, $S0$, PUP и $\overline{FE}$, можно представить 12 вариантов формирования адреса, реализуемых СУАМ за один такт (табл. 7.24). В табл. 7.24 буквенные обозначения регистров СУАМ, стоящие в квадратных скобках, обозначают содержимое этих регистров. Предполагается, что в исходном состоянии (на N-м такте) имеется: [СМК] — содержимое счетчика микрокоманд; [РА] — содержимое регистра адреса; [СТ0] ... [СТ3] — содержимое накопителя стека, причем СТ0 — регистр, адресуемый указателем стека (верхняя ячейка стека), остальные ячейки стека располагаются последовательно с возрастающими номерами СТ1, СТ2, СТ3. На вход $C0$ подана «1». На некоторых примерах этой таблицы и поясним формирование адресов в СУАМ.

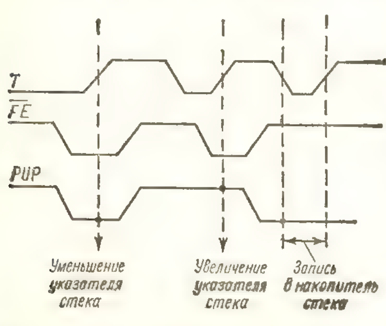


Рис. 7.26. Временна́я диаграмма управления стеком

Переход к следующей микрокоманде (№ 3). Сигналы $S1$, $S0$, равные «0», обеспечивают передачу на выход СУАМ текущего значения СМК в N-м такте. Сигнал $\overline{FE}$ = «1», следовательно стек отключен. Поэтому на очередном положительном фронте тактового сигнала, при переходе к $(N + 1)$-му такту, содержимое стека не изменится, к текущему значению СМК на входе СМК будет добавлена 1 и в регистр СМК запишется адрес [СМК] + 1.

Переход к адресу на D-входах (№ 12). При $S1$, $S0$, равных «1», на выход СУАМ в N-м такте передаются входы D, а так как $\overline{FE}$ = «1», то стек отключен и на (N + 1)-м такте в СМК содержится $[D] + 1$.

Засылка содержимого СМК в стек и переход к адресу на входах D (№ 11). Сигналы $S1$, $S0$ в исходном состоянии равны «1», поэтому, как и в предыдущем примере, на выходе Y в течение N-го такта будет $[D]$, а содержимое СМК на $(N + 1)$-м такте равно $[D] + 1$. Сигнал $\overline{FE}$ = «0», т. е. стек включен, а PUP = = «1» указывает, что выполняется операция записи информации в стек. В течение N-го такта на входе стека находится слово [СМК]

Таблица 7.24. **Операции СУАМ**

Вариант	Цикл	Сигналы на входах $S1$, $S0$, $\overline{FE}$, PUP	СМК	РА	Накопитель стека				Выход Y	Комментарий
					СТ0	СТ1	СТ2	СТ3		
1	N $N+1$	0000 —	[СМК] [СМК]+1	[РА] [РА]	[СТ0] [СТ1]	[СТ1] [СТ2]	[СТ2] [СТ3]	[СТ3] [СТ0]	[СМК] —	Выталкивание из стека
2	N $N+1$	0001 —	[СМК] [СМК]+1	РА РА	[СТ0] [СМК]	[СТ1] [СТ0]	[СТ2] [СТ1]	[СТ3] [СТ2]	[СМК] —	Засылка [СМК] в стек
3	N $N+1$	001X —	[СМК] [СМК]+1	РА РА	[СТ0] [СТ0]	[СТ1] [СТ1]	[СТ2] [СТ2]	[СТ3] [СТ3]	[СМК] —	Переход к следующей микрокоманде
4	N $N+1$	0100 —	[СМК] [РА]+1	РА РА	[СТ0] [СТ1]	[СТ1] [СТ2]	[СТ2] [СТ3]	[СТ3] [СТ0]	[РА] —	Выталкивание из стека; переход к адресу в РА
5	N $N+1$	0101 —	[СМК] [РА]+1	РА РА	[СТ0] [СМК]	[СТ1] [СТ0]	[СТ2] [СТ1]	[СТ3] [СТ2]	[РА] —	Засылка [СМК] в стек, переход к адресу в РА
6	N $N+1$	011X —	[СМК] [РА]+1	РА РА	[СТ0] [СТ0]	[СТ1] [СТ1]	[СТ2] [СТ2]	[СТ3] [СТ3]	[РА] —	Переход к адресу в РА
7	N $N+1$	1000 —	[СМК] [СТ0]+1	РА РА	[СТ0] [СТ1]	[СТ1] [СТ2]	[СТ2] [СТ3]	[СТ3] [СТ0]	[СТ0] —	Переход к адресу в СТ0, выталкивание из стека
8	N $N+1$	1001 —	[СМК] [СТ0]+1	РА РА	[СТ0] [СМК]	[СТ1] [СТ0]	[СТ2] [СТ1]	[СТ3] [СТ2]	[СТ0] —	Переход к адресу в СТ0, засылка [СМК] в стек
9	N $N+1$	101X —	[СМК] [СТ0]+1	РА РА	[СТ0] [СТ0]	[СТ1] [СТ1]	[СТ2] [СТ2]	[СТ3] [СТ3]	[СТ0] —	Переход к адресу в СТ0
10	N $N+1$	1100 —	[СМК] [D]+1	РА РА	[СТ0] [СТ1]	[СТ1] [СТ2]	[СТ2] [СТ3]	[СТ3] [СТ0]	[D] —	Переход к адресу на входах D; выталкивание из стека
11	N $N+1$	1101 —	[СМК] [D]+1	РА РА	[СТ0] [СМК]	[СТ1] [СТ0]	[СТ2] [СТ1]	[СТ3] [СТ2]	[D] —	Переход к адресу на входах D; засылка [СМК] в стек
12	N $N+1$	111X —	[СМК] [D]+1	РА РА	[СТ0] [СТ0]	[СТ1] [СТ1]	[СТ2] [СТ2]	[СТ3] [СТ3]	[D] —	Переход к адресу на входах D

На очередном положительном фронте тактового импульса в момент перехода к ($N + 1$)-му такту это слово будет записано в регистр записи адреса для временного запоминания, по этому же фронту увеличится указатель стека и отрицательным импульсом тактового сигнала T в ($N + 1$)-м такте слово из регистра записи адреса перепишется в верхнюю ячейку стека (в ячейку СТ0). Содержимое СТ0 переместится в СТ1 и т. д., причем старое содержимое самой нижней ячейки СТ3 вытолкнется из стека — потеряется. Физически обмена информацией между ячейками стека не происходит.

Выталкивание из стека и переход к адресу на входах D (№ 10). Эта операция отличается от предыдущей значением сигнала PUP (PUP = «0»), поэтому информация выталкивается из стека. Указатель стека уменьшается на 1, и бывшая верхняя ячейка становится нижней, при этом изменения информации в регистрах накопителя стека не происходит.

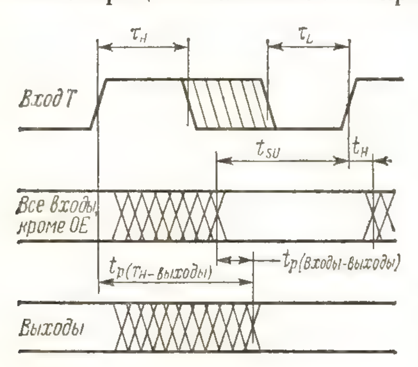


Рис. 7.27. Временна́я диаграмма работы СУАМ

Статические и динамические параметры микросхем К1804ВУ1 и К1804ВУ2 даны в табл. 7.25 и табл. 7.26 соответственно.

Временная диаграмма работы СУАМ показана на рис. 7.27.

Примеры применения. Используя микросхемы СУАМ1 и СУАМ2, можно получать различные структуры блока микропрограммного управления (БМУ). На рис. 7.28 показана простая структура БМУ на основе СУАМ К1804ВУ2. Кроме СУАМ в БМУ входят запоминающее устройство микрокоманд *ЗУМК* и регистр микрокоманд *РМК*. В микрокоманде выделены поле управления СУАМ и поле адреса ветвления, разряды которого поступают непосредственно на входы D СУАМ.

В качестве примера рассмотрим организацию перехода к подпрограмме с возвратом. Функционирование блока микропрограммного управления дано в табл. 7.27. В табл. 7.27, *а* условно обозначены два участка микропрограммы, причем адреса K ... ($K + 4$) относятся к основной программе, а адреса A ... ($A + 2$) — к подпрограмме. Микрокоманда, выбранная из *ЗУМК* по адресу K, обозначена как I (K), по адресу $K + 1$ — как I ($K + 1$) и т. д. В табл. 7.27, *б* описана последовательность выполнения отдельных микрокоманд в соответствии с тактовыми сигналами T. Содержимое *РМК* или *ЗУМК* по какому-либо адресу представлено, как значение адреса, в квадратных скобках.

В цикле $T0$ выполняют микрокоманду с адресом K. $S1$, $S0$ равны «0», поэтому на выход СУАМ из СМК поступает адрес

$K + 1$. На выходе *ЗУМК* — микрокоманда с адресом $K + 1$. В цикле $T1$ выполняют микрокоманду с адресом $K + 1$. $S1$, $S0$ равны «0», поэтому на выход СУАМ из СМК поступает адрес $K + 2$. По адресу $K + 2$ в ЗУМК находится микрокоманда, управляющая часть которой содержит инструкцию «Переход к подпрограмме A» (JSRA).

В цикле Т2 эта команда находится в *РМК* и на входах СУАМ устанавливаются сигналы для выполнения перехода и сохранения

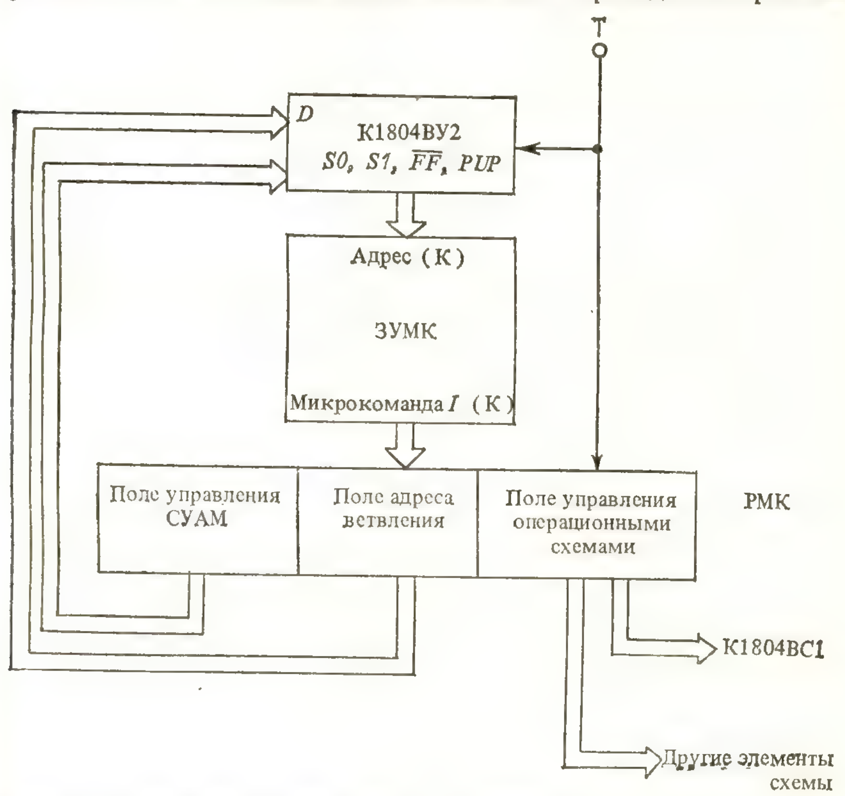


Рис. 7.28. Структурная схема БМУ на К1804ВУ2

адреса возврата. Адрес подпрограммы A передается из РМК на входы D СУАМ и поступает на выходы Y. По этому адресу из ЗУМК выбирается первая микрокоманда подпрограммы A и поступает на вход РМК. В цикле Т3 микрокоманда с адресом A загружается в РМК, а адрес возврата $K + 3$ из СМК засылается в стек. Адрес $A + 1$ из СУАМ поступает на вход *ЗУМК*. Команда с адресом $A + 1$ из *ЗУМК* поступает на вход *РМК*. В цикле Т4 адрес $A + 2$ передается на выходы СУАМ. В управляющей части команды с адресом $A + 2$ содержится инструкция «Возврат из подпрограммы» (RTS). В цикле Т5 на входах СУАМ устанавли-

Таблица 7.25. **Статические параметры микросхем К1804ВУ1, К1804ВУ2**

Параметр, единица измерения	Обозначение	Значение параметров		Режим измерения
		мин.	макс.	
Входной ток «0», мА, для входов: *T*, *RE*, *ZA*, *S*0, *S*1, *FE*, *R*3 ... *R*0, *OR*3 ... *OR*0, *D*3 ... *D*0 (К1804ВУ1) *OE*, *PUP*, *D*3 ... *D*0 (К1804ВУ2) *C*0	I_{IL}	— — —	—0,36 —0,72 —1,08	U_{CC} = 5,25 В U_{IL} = 0,4 В
Входной ток «1», мкА, для входов: *T*, *RE*, *ZA*, *S*0, *S*1, *FE*, *R*3 ... *R*0, *OR*3 ... *OR*0, *D*3 ... *D*0 (К1804ВУ1) *C*0, *PUP*, *D*3 ... *D*0 (К1804ВУ2)	I_{IH}	— —	20 40	U_{CC} = 5,25 В U_{IH} = 2,7 В
Выходное напряжение «0», В	U_{OL}	—	0,5 0,45 0,4	I_{OL} = 12 мА (для *Y*3 ... *Y*0) I_{OL} = 8 мА (для *C*4) I_{OL} = 4 мА (для *C*4)
Выходное напряжение «1», В	U_{OH}	2,4	—	I_{OH} = —1 мА
Прямое падение напряжения на антизвонном диоде, В	U_{CDI}	—	1,5	U_{CC} = 4,75 В I_I = —18 мА
Выходной ток в состоянии «выключено» при напряжении «0» на выходе (*Y*3 ... *Y*0), мкА	I_{OZL}	—	—20	U_{CC} = 5,25 В U_{OL} = 0,4 В
То же при напряжении «1», мкА	I_{OZH}	—	20	U_{CC} = 5,25 В U_{OH} = 2,7 В
Ток короткого замыкания, мА, для выходов: *Y*3 ... *Y*0 *C*4	I_{OS}	—30 —30	—100 —85	U_{CC} = 5,25 В U_{OL} = 0 В
Ток потребления, мА	I_{CC}	—	130	U_{CC} = 5,25 В

Таблица 7.26. **Динамические параметры микросхем К1804ВУ1 и К1804ВУ2**

Параметр	Значение, нс		Температура, °С
	мин.	макс.	
Длительность тактового сигнала низкого уровня τ_L на входе *T*	35	—	25
	40	—	—10 ... 70
Длительность тактового сигнала высокого уровня τ_H на входе *T*	35	—	25
	40	—	—10 ... 70
Время задержки распространения сигнала t_p			
от входов *D* до выходов *Y*	—	20	25
	—	25	—10 ... 70
от входов *D* до выходов *C*4	—	32	25
	—	37	—10 ... 70
от входов *S*0, *S*1 до выходов *Y*	—	40	25
	—	50	—10 ... 70
от входов *S*0, *S*1 до выходов *C*4	—	50	25
	—	65	—10 ... 70
от входов *OR* до выходов *Y* (для К1804ВУ1)	—	20	25
	—	25	—10 ... 70
от входов *OR* до выходов *C*4 (для К1804ВУ1)	—	32	25
	—	37	—10 ... 70
от входов *C*0 до выходов *C*4	—	16	25
	—	20	—10 ... 70
от входов *ZA* до выходов *Y*	—	40	25
	—	50	—10 ... 70
от входов *ZA* до выходов *C*4	—	50	25
	—	65	—10 ... 70
от входов *T* до выходов *Y* (при операции *S*1=0, *S*0=1)	—	50	25
	—	65	—10 ... 70
от входов *T* до выходов *Y* (при операции *S*1=0, *S*0=0)	—	50	25
	—	65	—10 ... 70
от входов *T* до выходов *Y* (при операции *S*1=1, *S*0=1)	—	62	25
	—	77	—10 ... 70
от входов *T* до выходов *Y* (при операции *S*1=1, *S*0=0)	—	90	25
	—	105	—10 ... 70
от входов *T* до выходов *C*4 (при операции *S*1=1, *S*0=0)	—	102	25
	—	112	—10 ... 70
Время установления сигнала t_{SU} относительно положительного фронта сигнала на входе *T* на входах:			
RE	22	—	25
	25	—	—10 ... 70
R (для К1804ВУ1)	12	—	25
	15	—	—10 ... 70
D (при записи в регистр адреса для К1804ВУ2)	12	—	25
	15	—	—10 ... 70
PUP	30	—	25
	35	—	—10 ... 70
FE	30	—	25
	35	—	—10 ... 70
*C*0	30	—	25
	35	—	—10 ... 70
D (при записи в счетчик микрокоманд)	35	—	25
	45	—	—10 ... 70

Параметр	Значение, нс		Температура, °С
	мин.	макс.	
OR (для К1804ВУ1)	35	—	25
	45	—	—10 ... 70
*S*0, *S*1	50	—	25
	60	—	—10 ... 70
ZA	50	—	25
	60	—	—10 ... 70
Время сохранения сигнала t_H относительно положительного перепада сигнала на входе *T* на входах:			
RE	5	—	25
	7	—	—10 ... 70
R (для К1804ВУ1)	5	—	25
	7	—	—10 ... 70
D (при записи в регистр адреса для К1804ВУ2)	5	—	25
	7	—	—10 ... 70
PUP	7	—	25
	10	—	—10 ... 70
FE, *C*0	5	—	25
	7	—	—10 ... 70
D (при записи в счетчик микрокоманд)	3	—	25
	8	—	—10 ... 70
OR (для К1804ВУ1)	3	—	25
	8	—	—10 ... 70
*S*0, *S*1	0	—	25
	2	—	—10 ... 70
ZA	0	—	25
	2	—	—10 ... 70
Время задержки распространения сигнала при переходе из состояния «выключено» в состояние «1» t_{ZH} от входа *OE* до выходов *Y*	—	25	25
	—	30	—10 ... 70
Время задержки распространения сигнала при переходе из состояния «выключено» в состояние «0» t_{ZL} от входа *OE* до выходов *Y*	—	25	25
	—	30	—10 ... 70
Время задержки распространения сигнала при переходе из состояния «1» в состояние «выключено» t_{HZ} от входа *OE* до выходов *Y*	—	25	25
	—	30	—10 ... 70
Время задержки распространения сигнала при переходе из состояния «0» в состояние «выключено» t_{LZ} от входа *OE* до выходов *Y*	—	25	25
	—	30	—10 ... 70

Примечание. Временные параметры измерялись при $U_{CC} = 5 \pm 0{,}25$ В и $C_L = 50$ пФ.

Таблица 7.27. Выполнение подпрограммы

а)

Выполняемый цикл	Микропрограмма: Адрес	Микропрограмма: Микрокоманда
	$K-1$	
T0	K	$I(K)$
T1	$K+1$	$I(K+1)$
T2	$K+2$	$JSRA$
T6	$K+3$	$I(K+3)$
T7	$K+4$	$I(K+4)$
	—	
	—	
	—	
T3	A	$I(A)$
T4	$A+1$	$I(A+1)$
T5	$(A+2)$	RTS
	—	—

Примечание. $C0$ = «1».

б)

Цикл выполнения		T0	T1	T2	T3	T4	T5	T6	T7
Входы СУАМ (от РМК)	$S1, S0$ $\overline{FE}$ PUP D	00 1 X X	00 1 X X	11 0 1 A	00 1 X X	00 1 X X	10 0 0 X	00 1 X X	00 1 X X
Внутренние регистры СУАМ	СМК СТ0 СТ1 СТ2 СТ3	$K+1$ — — — —	$K+2$ — — — —	$K+3$ — — — —	$A+1$ $K+3$ — — —	$A+2$ $K+3$ — — —	$A+3$ $K+3$ — — —	$K+4$ — — — —	$K+5$ — — — —
Выход СУАМ	Y	$K+1$	$K+2$	A	$A+1$	$A+2$	$K+3$	$K+4$	$K+5$
Выход ЗУМК	$[Y]$	$[K+1]$	$JSRA$	$[A]$	$[A+1]$	RTS	$[K+3]$	$[K+4]$	$[K+5]$
Содержимое РМК (выполняемая микрокоманда)	[РМК]	$[K]$	$[K+1]$	$JSRA$	$[A]$	$[A+1]$	RTS	$[K+3]$	$[K+4]$

ваются сигналы для выполнения возврата из подпрограммы. Из стека СУАМ извлекается адрес возврата $K+3$ и поступает на выходы Y. По адресу $K+3$ из *ЗУМК* считывается микрокоманда основной микропрограммы и поступает на вход РМК. С цикла Т6 продолжается выполнение основной микропрограммы (микрокоманды с адресами $K+3$, $K+4$ и т. д.).

В табл. 7.28 показано, как блок микропрограммного управления обеспечивает выполнение двух вложенных подпрограмм (A и B), причем подпрограмма B для простоты примера содержит всего одну микрокоманду.

Циклы Т0 ... Т3 работы СУАМ и блока микропрограммного управления аналогичны предыдущему примеру.

В цикле Т4 адрес $A+2$ передается на выходы СУАМ. В управляющей части команды с адресом $A+2$ содержится команда «Переход к подпрограмме B» (JSRB). Выполняется команда с адресом $A+1$.

В цикле Т5 выполняется команда JSRB, адрес B поступает из РМК на входы D СУАМ и передается на выходы Y. По этому адресу из ЗУМК считывается единственная команда подпрограммы B и передается на входы РМК. В цикле Т6 на входах СУАМ устанавливаются сигналы для выполнения возврата из подпрограммы B. В это время адрес возврата из подпрограммы B ($A+3$) запи-

Таблица 7.28. **Выполнение двух вложенных подпрограмм**

а)

Выполняемый цикл	Микропрограмма: Адрес	Микропрограмма: Микрокоманда
	$K-1$	
Т0	K	$I(K)$
Т1	$K+1$	$I(K+1)$
Т2	$K+2$	JSRA
Т9	$K+3$	$I(K+3)$
	—	
	—	
	—	
	—	
Т3	A	$I(A)$
Т4	$A+1$	$I(A+1)$
Т5	$A+2$	JSRB
Т7	$A+3$	$I(A+3)$
Т8	$A+4$	RTS
	—	
	—	
Т6	B	RTS
	—	
	—	

Примечание. $C0 =$ «1»

б)

Цикл выполнения		Т0	Т1
Входы СУАМ (от РМК)	$S1, S0$ FE PUP D	00 1 X X	00 1 X X
Внутренние регистры СУАМ	СМК СТ0 СТ1 СТ2 СТ3	$K+1$ — — — —	$K+2$ — — — —
Выход СУАМ	Y	$K+1$	$K+2$
Выход ЗУМК	$[Y]$	$[K+1]$	JSRA
Содержимое РМК (выполняемая микрокоманда)	[РМК]	$[K]$	$[K+1]$

сывается в верхнюю ячейку стека (СТ0), а адрес возврата из подпрограммы A ($K+3$) «протолкнут» в следующую ячейку стека (СТ1). Из стека СУАМ считывается адрес возврата $A+3$ и поступает на выходы Y. По адресу $A+3$ из ЗУМК считывается следующая команда подпрограммы A и поступает на вход РМК. В цикле Т7 выполняется команда с адресом $A+3$. Из СУАМ на вход ЗУМК поступает адрес $A+4$ и считывается соответствующая команда. Это команда возврата из подпрограммы A. В цикле Т8 выполняется команда возврата (RTS) из подпрограммы A. На входах СУАМ устанавливаются сигналы, обеспечивающие считывание из стека адреса возврата $K+3$. Этот адрес с выходов СУАМ поступает на вход ЗУМК. Происходит считывание команды с адресом $K+3$ и передача ее на вход РМК. С цикла Т9 продолжается выполнение микропрограммы (команды $K+3$, $K+4$ и др.).

СУАМ позволяет выполнить четыре вложенные друг в друга подпрограммы, что обеспечивает достаточно большую гибкость микропрограммного управления, построенного на СУАМ.

Наращивание СУАМ. Для формирования адресов, разрядность которых больше четырех, необходимо объединение нескольких СУАМ: например, блок из трех схем позволяет обращаться к памяти объемом в 4К слов.

T2	T3	T4	T5	T6	T7	T8	T9
11 0 1 A	00 1 X X	00 1 X X	11 0 1 B	10 0 0 X	00 1 X X	10 0 0 X	00 1 X X
$K+3$ — — — —	$A+1$ $K+3$ — — —	$A+2$ $K+3$ — — —	$A+3$ $K+3$ — — —	$B+1$ $A+3$ $K+3$ — —	$A+4$ $K+3$ — — —	$A+5$ $K+3$ — — —	$K+4$ — — — —
A	$A+1$	$A+2$	B	$A+3$	$A+4$	$K+3$	$K+4$
$[A]$	$[A+1]$	JSRB	RTS	$[A+3]$	RTS	$[K+3]$	$[K+4]$
JSRA	$[A]$	$[A+1]$	JSRB	RTS	$[A+3]$	*RTS*	$[K+3]$

Пример объединения трех схем К1804ВУ2 приведен на рис. 7.29. При наращивании необходимо соединить не только шины управления (*S*0, *S*1, *FE*, *PUP*, Т, *RE*, *OE*) различных каскадов, но и выход переноса *C*4 предыдущего каскада со входом переноса *C*0 последующего.

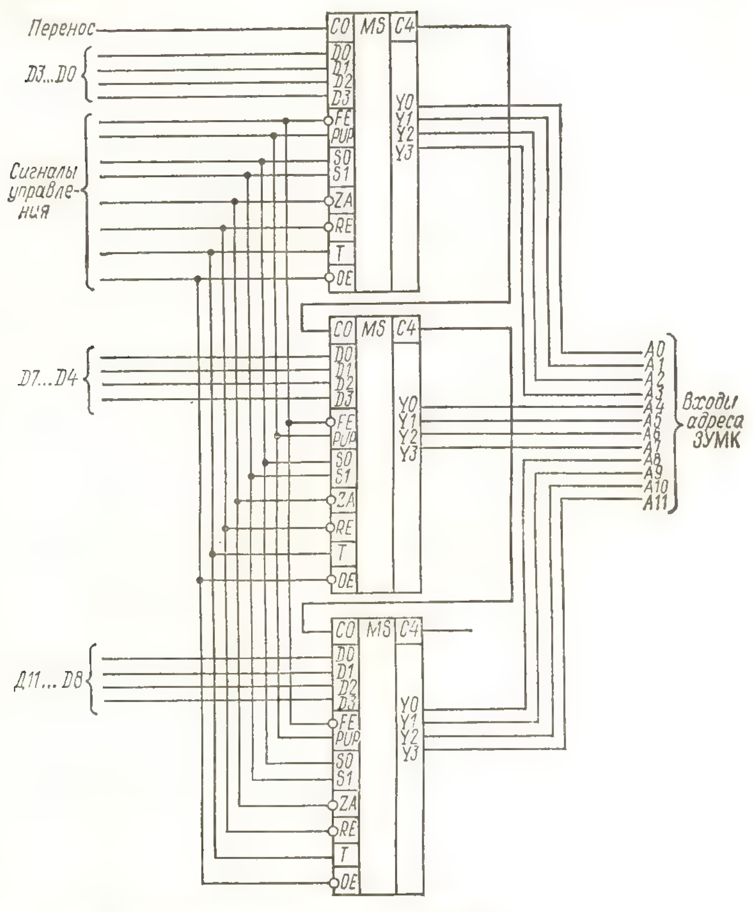
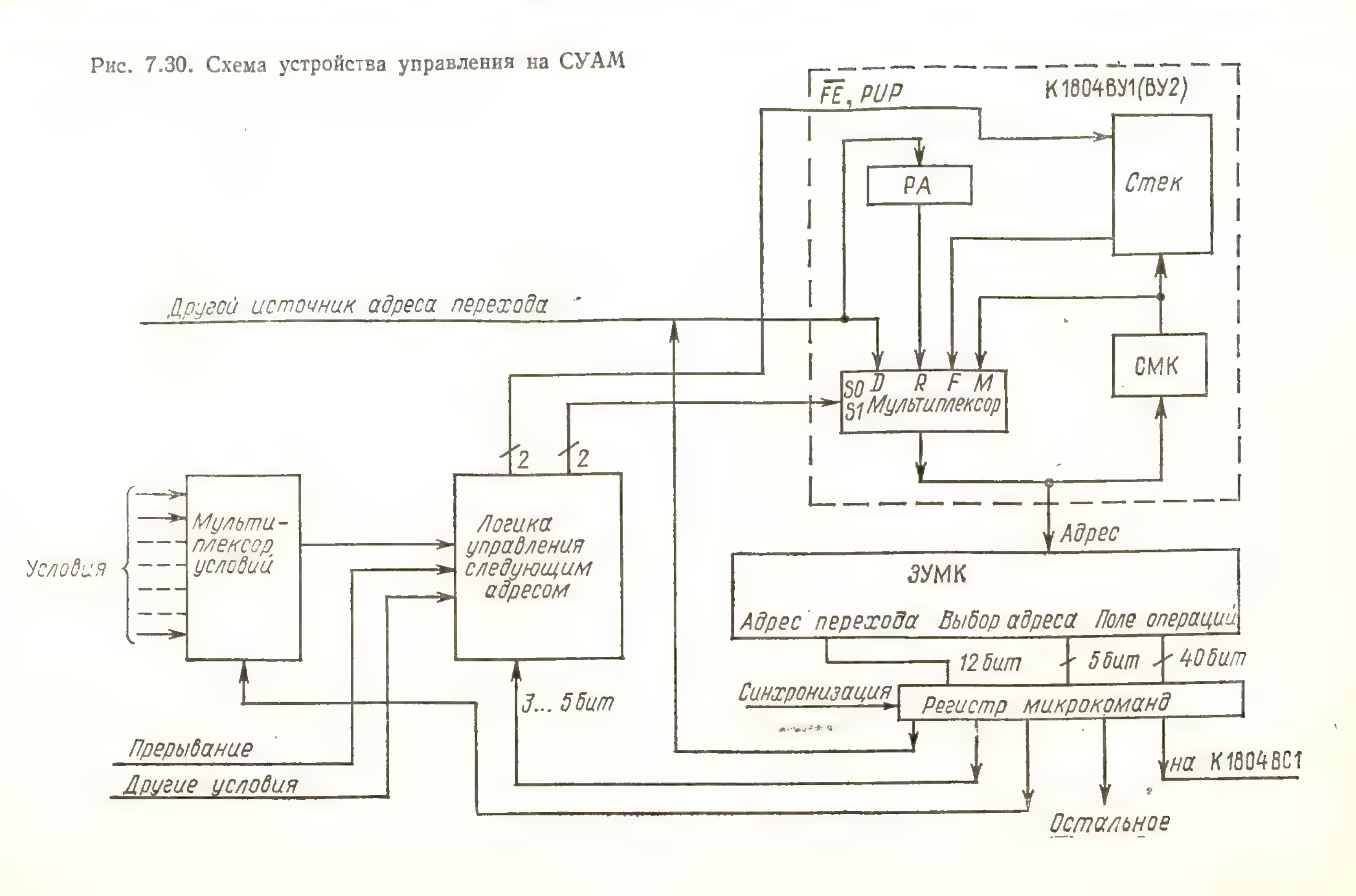


Рис. 7.29. Схема объединения трех К1804ВУ2

Сигнал переноса *C*0 обладает управляющими функциями, поэтому при выборе параметров тактовой последовательности необходимо обеспечить время, достаточное для установления *C*0 на входе старшей СУАМ до момента перехода тактового сигнала с уровня «0» на уровень «1». Это также следует учитывать в случае, когда адрес с СУАМ поступает на регистр адреса и сигнал переноса должен установиться до подачи тактового импульса на регистр. Если адрес с СУАМ поступает на управляющую память,

Рис. 7.30. Схема устройства управления на СУАМ



то прохождение сигнала переноса через СУАМ происходит во время обращения к памяти.

Архитектура устройства управления на СУАМ. На рис. 7.30 приведена рекомендуемая архитектура, использующая К1804ВУ2 (или К1804ВУ1). Адрес из СУАМ поступает на запоминающее устройство микрокоманд *ЗУМК*. *Регистр микрокоманд* содержит выполняемую микрокоманду. Часть этой микрокоманды предназначена для управления операционными блоками, в частности

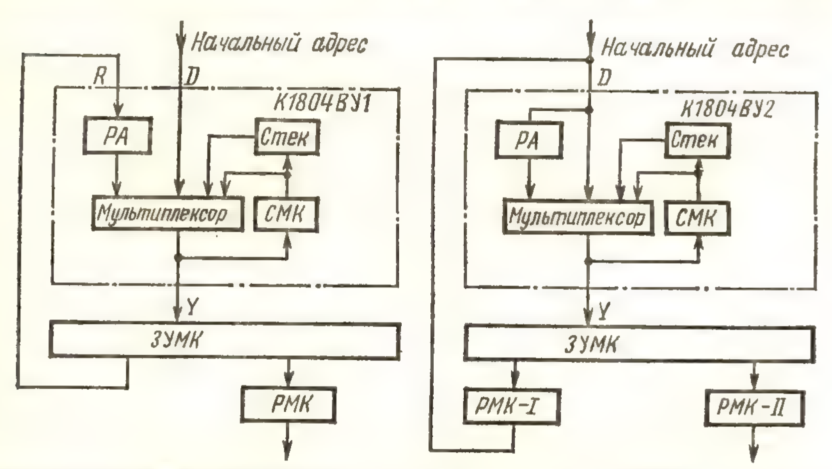


Рис. 7.31. Структуры формирования адресов переходов на К1804ВУ1 и К1804ВУ2

микропроцессорными секциями К1804ВС1. Другая часть этой микрокоманды содержит команду управления, например такую, как «Продолжение», «Цикл», «Возврат из подпрограммы» и т. д. Биты, которые представляют эту команду, логически комбинируются в блоке *Логика управления следующим адресом* с битами, представляющими условия проверки и состояния системы, выбор которых с помощью мультиплексора условий также управляется от разрядов микрокоманды. Сигналы с блока *логика управления следующим адресом* управляют СУАМ, обеспечивая подготовку адреса следующей микрокоманды. Шину входов *D* СУАМ используют для подачи адреса перехода как от разрядов микрокоманды, так и от других источников.

Блок *логика управления следующим адресом* может состоять из обычных вентилей, ПЗУ или программируемой логической матрицы. В качестве этого блока может быть использована микросхема рассматриваемой серии К1804ВУЗ. Ее применение описано в разд. 7.5.

Использование СУАМ1 и СУАМ2. В СУАМ1 имеются отдельные входы для регистра хранения и непосредственные входы в мультиплексор. В СУАМ2 эти входы объединены, что может влиять на разработку структуры формирования адреса перехода

(рис. 7.31). При использовании СУАМ1 выходы *ЗУМК* могут быть соединены с входами регистра адреса *РА* СУАМ1. В этом случае *РА* заменяет часть регистра микрокоманд *РМК*, а входы *D* используются независимо. При использовании СУАМ2 на входы D объединяются различные источники адресов. Внутренний регистр *РА* применяется лишь для временного хранения некоторых предыдущих адресов переходов. В регистре микрокоманд необходимы дополнительные разряды (*РМК*-1).

Для построения блоков с расширенными возможностями рекомендуется использовать микросхему СУАМ1 (К1804ВУ1), которая имеет входы маски *OR*3 ... *OR*0. На входы маски подается одно или несколько маскирующих условий. Обычно выбирают переход по адресу с нулями в младших битах. Эти биты заменяют результатом выполнения операции ИЛИ над выходными сигналами мультиплексора СУАМ1 и сигналами маски. На рис. 7.32 результатом состояния маскирующих условий *X* и *Y* является переход по адресам 1100, 1101, 1110, 1111.

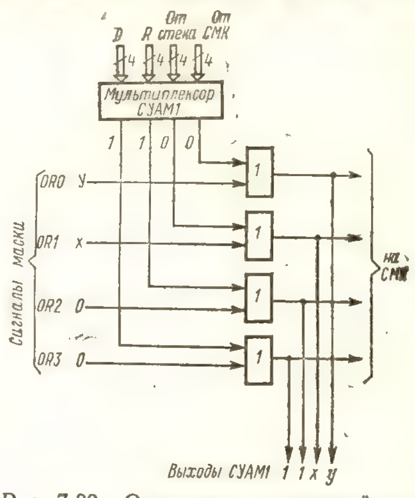


Рис. 7.32. Организация ветвлений по входам маски

7.5. СХЕМА УПРАВЛЕНИЯ СЛЕДУЮЩИМ АДРЕСОМ К1804ВУ3

Схема управления следующим адресом (УСА) предназначена для формирования управляющих сигналов для устройств блока микропрограммного управления, построенного на базе СУАМ, в зависимости от четырехразрядного поля микрокоманды и входа признака ветвления. Все входные сигналы управления СУАМ вырабатываются в микросхеме УСА.

Условное графическое обозначение УСА представлено на рис. 7.33, назначение выводов описано в табл. 7.29.

Функциональная схема УСА показана на рис. 7.34. Схема в целом представляет собой комбинационный преобразователь на основе ПЗУ объемом 32×8 бит. Входы *I*3 ... *I*0 и *TST* фактически являются адресными входами. Выходы микросхемы представляют собой буферы с тремя устойчивыми состояниями, работой которых управляет вход *OE*. Когда на вход *OE* подана «1», то выходные буфера имеют состояние высокого сопротивления и отключены от шин. Если же на вход *OE* подан «0» ,то выходные буфера передают сигналы, сформированные микросхемой. Так как микросхема

К1804ВУ3 является чисто комбинационной, то на подачу сигналов микрокоманды (*I*3 ... *I*0) и сигнала признака ветвления (*TST*) временных ограничений не накладывается. Схема УСА позволяет реализовать 16 инструкций, управляющих последовательностью микрокоманд. Типы этих инструкций в зависимости от входных сигналов микрокоманды, подаваемых на УСА, и мнемонические обозначения инструкций представлены в табл. 7.30. Соответствие значений входных и выходных сигналов приведено в таблице истинности (табл. 7.31).

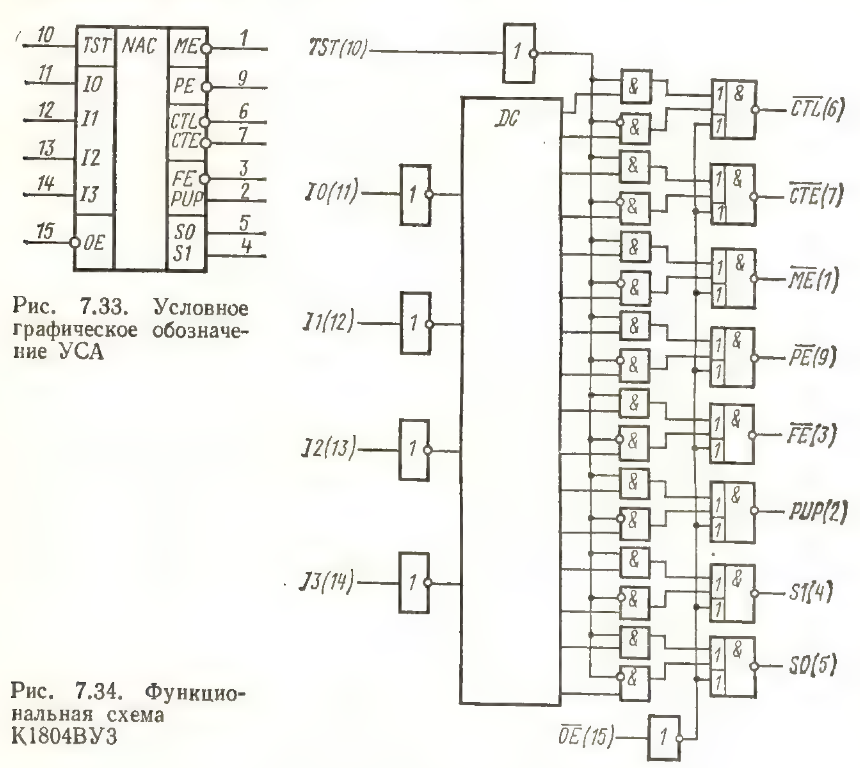


Рис. 7.33. Условное графическое обозначение УСА

Рис. 7.34. Функциональная схема К1804ВУ3

Статические и динамические параметры УСА К1804ВУ3 представлены в табл. 7.32 и табл. 7.33 соответственно.

Рассмотрим особенности применения УСА в составе блока микропрограммного управления. УСА рассчитана в основном на применение совместно с СУАМ К1804ВУ1, К1804ВУ2 и, хотя выходные сигналы УСА можно использовать по-разному, рекомендуется типовая структура БМУ (рис. 7.35) на базе *СУАМ* и *УСА*. Центральной частью БМУ является *СУАМ*, построенная в данном случае на К1804ВУ2. Адрес микрокоманды с выхода *СУАМ* подается на адресный вход *ЗУМК*. Считанная из *ЗУМК* микрокоманда записывается в *Регистр микрокоманд* (РМК). Микро-

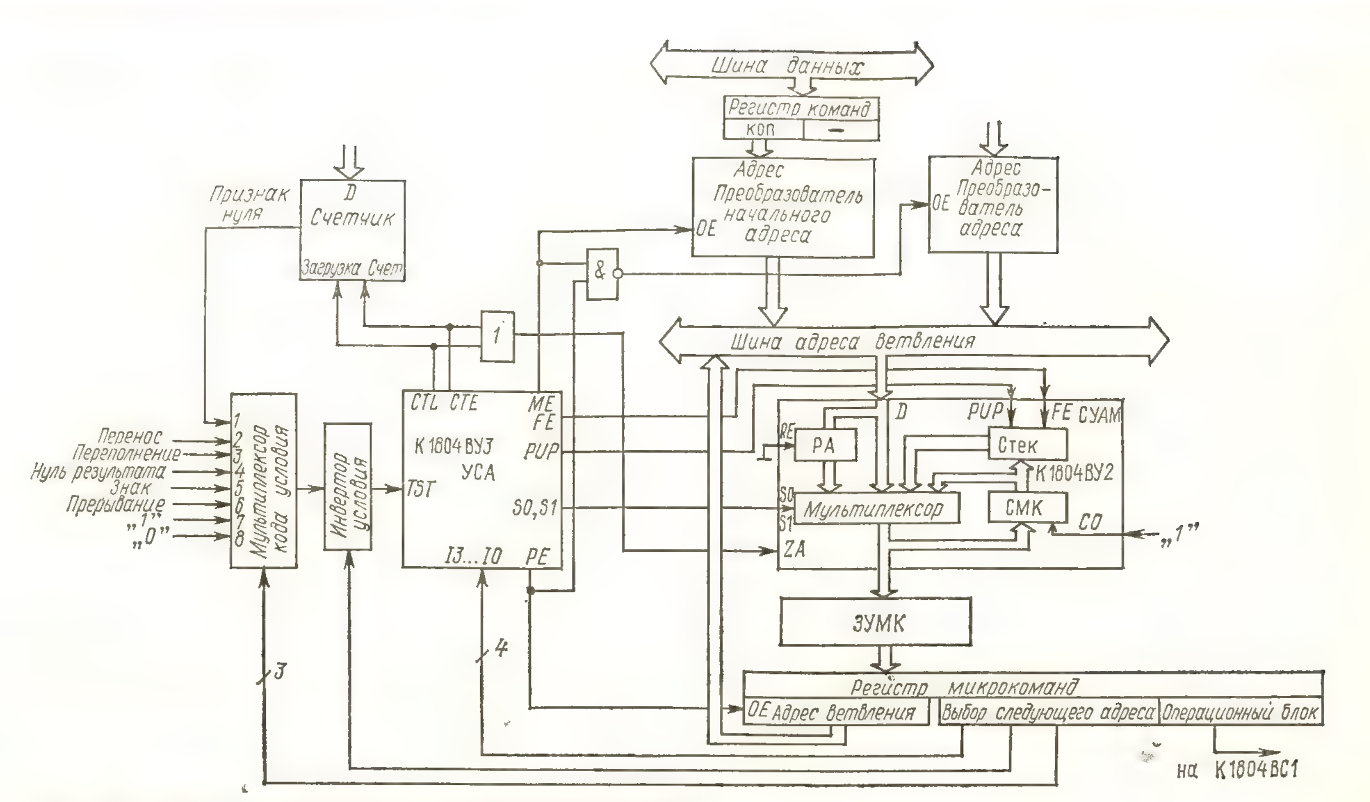


Рис. 7.35. Типовая структура БМУ на К1804ВУ2 и К1804ВУ3

Таблица 7.29. **Назначение выводов микросхемы К1804ВУ3**

Номер вывода	Назначение	Обозначение	Разряд	Тип
1	Разрешение работы ПЛМ	ME	—	Выход
2	Управление стеком	PUP	—	Выход
3	Разрешение управления стеком	FE	—	Выход
4	Выбор адреса	$S1$	1	Выход
5	Выбор адреса	$S0$	0	Выход
6	Разрешение загрузки	CTL	—	Выход
7	Разрешение счета	CTE	—	Выход
8	Общий	GND	—	—
9	Разрешение для регистра микрокоманд	PE	—	Выход
10	Признак ветвления	TST	—	Вход
11	Микрокоманда	$I0$	0	Вход
12	Микрокоманда	$I1$	1	Вход
13	Микрокоманда	$I2$	2	Вход
14	Микрокоманда	$I3$	3	Вход
15	Разрешение выходов	OE	—	Вход
16	Питание	U_{CC}	—	—

Примечание. Все выходы трехстабильные.

команда имеет ряд управляющих полей, два из которых *Адрес ветвления* и *Выбор следующего адреса* предназначены для самого блока микропрограммного управления. Биты поля *Операционный блок* могут управлять непосредственно микропроцессорной секцией К1804ВС1.

Текущая команда с шины данных поступает на *Регистр команд*. Разряды, определяющие код операции *КОП*, поступают на *Преобразователь начального адреса*, который может быть выполнен в виде ПЗУ или программируемой логической матрицы (ПЛМ). *Преобразователь начального адреса* преобразует код операции в адрес первой микрокоманды в микропрограмме, соответствующей данной команде. Этот адрес, при наличии разрешающего сигнала на входе *OE* преобразователя начального адреса, может быть передан на шину адреса ветвления (ШАВ), представляющую собой внутреннюю магистраль блока микропрограммного управления. Приемником информации с этой магистрали является *СУАМ* через прямые входы *D*. Вторым источником информации для ШАВ может служить часть РМК, содержащая поле адреса ветвления. Предполагается, что эта часть РМК имеет трехстабильные выходы, управляемые по входу *OE*. Третьим источником информации для ШАВ служит дополнительный *преобразователь адреса*, выполненный, так же как и *преобразователь начального адреса*, в виде ПЗУ или ПЛМ. Преобразователь адреса может содержать, например, адреса векторов прерывания. Подключение выхода преобразователя адреса к ШАВ также управляется по соответствующему входу разрешения *OE*.

Таблица 7.30. **Инструкции микросхемы К1804ВУ3**

Десятичное значение кода	Мнемоника	Входы				Инструкция
		I3	*I2*	*I1*	*I0*	
0	JZ	0	0	0	0	Перейти на нулевой адрес
1	CJS	0	0	0	1	Условный переход к подпрограмме на адрес из РМК
2	JMAP	0	0	1	0	Переход на адрес с выхода преобразователя начального адреса
3	CJP	0	0	1	1	Условный переход на адрес из РМК
4	PUSH	0	1	0	0	Засылка в стек и условная загрузка счетчика
5	JSRP	0	1	0	1	Условный переход к подпрограмме на адрес из РА либо из РМК
6	CJV	0	1	1	0	Условный переход на адрес вектора
7	JRP	0	1	1	1	Условный переход на адрес из РА либо из РМК
8	RFCT	1	0	0	0	Повторить цикл, если счетчик не равен нулю
9	RPCT	1	0	0	1	Повторить адрес из РМК, если счетчик не равен нулю
10	CRTN	1	0	1	0	Условный возврат из подпрограммы
11	CJPP	1	0	1	1	Условный переход на адрес из РМК и прием из стека
12	LDCT	1	1	0	0	Загрузить счетчик и продолжить
13	LOOP	1	1	0	1	Проверка конца цикла
14	CONT	1	1	1	0	Продолжить (перейти на следующий адрес)
15	JP	1	1	1	1	Безусловный переход на адрес из РМК

Таблица 7.31. Таблица истинности микросхемы К180УВУ3

Код (десятичное значение)	Инструкция	Входы					Выходы							
		I3	*I2*	*I1*	*I0*	*TST*	*S1*	*S0*	*FE*	*PUP*	*CTL*	*CTE*	*ME*	*PE*
0	JZ	0	0	0	0	X	1	1	1	1	0	0	1	0
1	CJS	0	0	0	1	0	0	0	1	1	1	1	1	0
						1	1	1	0	1	1	1	1	0
2	JMAP	0	0	1	0	X	1	1	1	1	1	1	0	1
3	CJP	0	0	1	1	0	0	0	1	1	1	1	1	0
						1	1	1	1	1	1	1	1	0
4	PUSH	0	1	0	0	0	0	0	0	1	1	1	1	0
						1	0	0	0	1	0	1	1	0
5	JSRP	0	1	0	1	0	0	1	0	1	1	1	1	0
						1	1	1	0	1	1	1	1	0
6	CJV	0	1	1	0	0	0	0	1	1	1	1	1	1
						1	1	1	1	1	1	1	1	1
7	JRP	0	1	1	1	0	0	1	1	1	1	1	1	0
						1	1	1	1	1	1	1	1	0
8	RFCT	1	0	0	0	0	1	0	1	0	1	0	1	0
						1	0	0	0	0	1	1	1	0
9	RPCT	1	0	0	1	0	1	1	1	1	1	0	1	0
						1	0	0	1	1	1	1	1	0
10	CRTN	1	0	1	0	0	0	0	1	0	1	1	1	0
						1	1	0	0	0	1	1	1	0
11	CJPP	1	0	1	1	0	0	0	1	0	1	1	1	0
						1	1	1	0	0	1	1	1	0
12	LDCT	1	1	0	0	X	0	0	1	1	0	1	1	0
13	LOOP	1	1	0	1	0	1	0	1	0	1	1	1	0
						1	0	0	0	0	1	1	1	0
14	CONT	1	1	1	0	X	0	0	1	1	1	1	1	0
15	JP	1	1	1	1	X	1	1	1	1	1	1	1	0

Таблица 7.32. **Статические параметры микросхемы К1804ВУ3**

Параметр, единица измерения	Обозначение	Значения параметров		Режим измерения
		мин.	макс.	
Входной ток «0», мА	I_{IL}	—	—0,25	U_{CC} = 5,25 В U_{IL} = 0,45 В
Входной ток «1», мкА	I_{IH}	—	25	U_{CC} = 5,25 В U_{IH} = 2,7 В
Выходное напряжение «0», В	U_{OL}	—	0,45	U_{CC} = 4,75 В U_{TL} = 0,8 В U_{IH} = 4,75 В U_{IL} = 0 В I_{OL} = 16 мА
Выходное напряжение «1», В	U_{OH}	2,4	—	U_{CC} = 4,75 В U_{TH} = 2,0 В U_{IH} = 4,75 В U_{IL} = 0 В I_{OH} = —2 мА
Прямое падение напряжения на антизвонном диоде, В	U_{CDI}	—	1,2	U_{CC} = 4,75 В I_I = —18 мА
Выходной ток в состоянии «выключено» при напряжении «0» на выходе, мкА	I_{OZL}	—	—40	U_{CC} = 5,25 В U_{OL} = 0,4 В
Выходной ток в состоянии «выключено» при напряжении «1» на выходе, мкА	I_{OZH}	—	40	U_{CC} = 5,25 В U_{OH} = 2,4 В
Ток короткого замыкания, мА	I_{OS}	—20	—90	U_{CC} = 5,25 В U_{OL} = 0 В
Ток потребления, мА	I_{CC}	—	115	U_{CC} = 5,25 В

Таблица 7.33. **Динамические параметры микросхемы К1804ВУ3**

Параметр	Максимальное значение, нс	Температура, °С
Время задержки распространения сигналов t_p от входов $I3$ ... $I0$ и TST до выходов	60 75	25 —10 ... 70
Время задержки распространения сигнала при переходе из состояния «выключено» в состояние «1» t_{ZH} от входа ОЕ до выходов	38 48	25 —10 ... 70
То же, в состояние «0» t_{ZL}	38 48	25 —10 ... 70
То же, из состояния «1» в состояние «выключено» t_{HZ}	30 40	25 —10 ... 70
То же, из состояния «0» в состояние «выключено t_{LZ}	30 40	25 —10 ... 70

Примечание. Временные параметры измерялись при $U_{CC} = 5 \pm 0{,}25$ В и $C_L = 50$ пФ.

В схеме БМУ предусмотрен счетчик, загружаемый по входу D от произвольного источника. Предполагается, что счетчик имеет выход признака, логически связанного каким-либо образом с его содержимым, например, признак нуля. Управление счетчиком выполняется по двум шинам активными нулевыми уровнями. Если на управляющем входе «Загрузка» «0», то выполняется загрузка счетчика по входу D; если на управляющем входе «Счет» «0», то с частотой синхронизирующей последовательности (она на схеме БМУ не указана) выполняется последовательное уменьшение на единицу содержимого счетчика. Данные, загружаемые в счетчик по входу D, могут поступать из любого источника, в том числе и с ШАВ.

Мультиплексор кода условия выбирает источник условия в зависимости от состояния соответствующих разрядов РМК. Источниками условия могут быть, например, данные с регистра состояния процессора: *перенос*, *переполнение*, *нуль результата*, *знак*, *признак нуля* от счетчика БМУ. В схеме БМУ (см. рис. 7.35). использован мультиплексор на восемь входов, но, очевидно, число входов может быть произвольным. С выхода *мультиплексора кода условия* информация передается на управляемый соответствующим разрядом РМК *инвертор условия*, позволяющий формировать инверсию от содержимого источника условия (например, условие «равно 0» заменять на условие «не равно 0» и т. п.). Сигнал TST с выхода *инвертора условия* и четыре управляющих сигнала $I3$ ... $I0$ с соответствующего поля РМК определяют работу схемы $УСА$. Выходные сигналы $УСА$ управляют остальными узлами блока микропрограммного управления: $СУАМ$ (сигналы $\overline{FE}$, PUP, $S0$, $S1$), счетчиком (сигналы $\overline{CTL}$, $\overline{CTE}$), преобразователями адресов (сигналы $\overline{ME}$, $\overline{PE}$), регистром микрокоманд(сигнал $\overline{PE}$).

В табл. 7.34 для всех 16 инструкций и сигнала ветвления представлены состояния счетчика циклов, стека и тип источника следующего адреса *СУАМ* как результат управляющих сигналов от *УСА*.

Применение *УСА* в приведенной структуре БМУ позволяет полностью реализовать все шестнадцать инструкций управления последовательностью микрокоманд (табл. 7.34). Ниже дается подробное описание этих инструкций при работе *УСА* в рекомендуемой структуре БМУ (см. рис. 7.35). Поясняющая информация приведена в табл. 7.35.

Инструкция JZ. Выполняется безусловный переход из любой точки микропрограммы к нулевому адресу — началу микропрограммы. Реализуется путем формирования на входе *ZA СУАМ* сигнала «0», как логической функции ИЛИ сигналов $\overline{CTL}$ и $\overline{CTE}$, равных «0» одновременно.

Инструкция CJS. При TST = «1» выполняется переход к адресу подпрограммы, принимаемому из РМК. Этот адрес (адрес M) устанавливается на входе *D СУАМ*. Адрес возврата ($N + 1$) запоминается в стеке. При TST = «0» выполняется следующая по порядку микрокоманда.

Инструкция JMAP. Выполняется безусловный переход к адресу, источником которого является *преобразователь начального адреса*. Этот адрес устанавливается на входе *D СУАМ*. Инструкция реализуется путем формирования на входе *OE* преобразователя начального адреса сигнала «0». Эта инструкция обычно завершает фазу вызова команды в БМУ, после чего начинается выполнение микропрограммы, соответствующей вызванной команде.

Инструкция CJP. При TST = «1» выполняется переход по адресу, принимаемому из РМК (адрес M), который устанавливается на входе *D* СУАМ. В отличие от инструкции CJS операция со стеком не производится. При TST = «0» выполняется следующая по порядку микрокоманда.

Инструкция PUSH. При выполнении текущей микрокоманды адрес следующей засылается в стек, одновременно с этим при TST = «1» формируется команда загрузки счетчика. При TST = «0» загрузка счетчика не производится. Естественный порядок следования микрокоманд не нарушается, и выполняется переход к следующей по порядку микрокоманде.

Инструкция JSRP. При выполнении инструкции прежде всего в стек засылается адрес возврата ($N + 1$). Далее при TST = «1» выполняется переход к адресу, источником которого является РМК (адрес M). Этот адрес устанавливается на входе *D* СУАМ. При TST = «0» выполняется переход к адресу P, источником которого является внутренний регистр РА СУАМ. Адрес P должен быть предварительно загружен в *РА*.

Инструкция CJV. При TST = «1» выполняется переход на адрес вектора, источником которого является дополнительный преобразователь адреса. Адрес V из этого преобразователя

Таблица 7.34. Состояния устройств СУАМ при выполнении инструкций УСА

Инструкция	Входы					Выходы				
	I3	I2	I1	I0	TST	Источник следующего адреса ($S0$, $S1$)	Стек ($\overline{FE}$, PUP)	Счетчик ($\overline{CTL}$, $\overline{CTE}$)	$\overline{ME}$	$\overline{PE}$
JZ	0	0	0	0	X	*D*	Хранение	X	1	0
CJS	0	0	0	1	0 1	СМК *D*	Хранение Засылка в стек	Хранение Хранение	1 1	0 0
JMAP	0	0	1	0	X	*D*	Хранение	Хранение	0	1
CJP	0	0	1	1	0 1	СМК *D*	Хранение Хранение	Хранение Хранение	1 1	0 0
PUSH	0	1	0	0	0 1	СМК СМК	Засылка в стек Засылка в стек	Хранение Загрузка	1 1	0 0
JSRP	0	1	0	1	0 1	РА *D*	Засылка в стек Засылка в стек	Хранение Хранение	1 1	0 0
CJV	0	1	1	0	0 1	СМК *D*	Хранение Хранение	Хранение Хранение	1 1	1 1
JRP	0	1	1	1	0 1	РА *D*	Хранение Хранение	Хранение Хранение	1 1	0 0
RFCT	1	0	0	0	0 1	СТЕК СМК	Хранение Извлечение из стека	Декремент Хранение	1 1	0 0
RPCT	1	0	0	1	0 1	*D* СМК	Хранение Хранение	Декремент Хранение	1 1	0 0
CRTN	1	0	1	0	0 1	СМК СТЕК	Хранение Извлечение из стека	Хранение Хранение	1 1	0 0
CJPP	1	0	1	1	0 1	СМК *D*	Хранение Извлечение из стека	Хранение Хранение	1 1	0 0
LDCT	1	1	0	0	X	СМК	Хранение	Загрузка	1	0
LOOP	1	1	0	1	0 1	СТЕК СМК	Хранение Извлечение из стека	Хранение Хранение	1 1	0 0
CONT	1	1	1	0	X	СМК	Хранение	Хранение	1	0
JP	1	1	1	1	X	*D*	Хранение	Хранение	1	0

Таблица 7.35. Выполнение операций в БМУ, построенном из схем УСА и СУАМ

Переход к нулевому адресу (JZ) 0 N	Условный переход к подпрограмме (CJS) TST = 0 TST = 1 M N N+1	Переход по адресу из преобразователя начального адреса (JMAP) M N
Условный переход по адресу из РМК (CJP) TST = 0 N TST = 1 M N+1	Засылка в стек и условная загрузка счетчика (PUSH) N ----- СТО: =N+1 N+1	Условный переход к одной из двух подпрограмм (JSRP) P TST=0 TST=1 M N N+1
Условный переход на адрес вектора (CJV) TST = 0 N TST = 1 V N+1	Переход на адрес, условно выбираемый из РМК либо РА (JRP) TST = 0 TST = 1 P N M	Повторный переход, если счетчик не равен нулю (RFCT) M TST = 1 N TST = 0 N+1
Условный переход к адресу из стека, если счетчик не равен нулю (RFCT) TST = 0 TST = 1 N M N+1	Повторять адрес из РМК, пока счетчик не равен нулю (RPCT) M TST = 1 N TST = 0 N+1	Условный возврат из подпрограммы (CRTN) N N+1 TST = 1 M M+1 TST = 0
Условный переход к адресу РМК и выталкивание из стека (CJPP) а TST = 1 TST = 0 N M N + 1 б N – 2 N – 1 TST = 1 TST1 = 0 N TST = 1 M1 TST2 = 0 N + 1 M2 TST3 = 0 N + 2 M3 N + 3 TST = 1 N + 4	Загрузка счетчика и продолжение (LDCT) N N+1	Контроль конца цикла (LOOP) TST = 0 TST = 1 N N+1
	Продолжить (CONT) N N+1	Безусловный переход к адресу из РМК (JP) N M

устанавливается на входе D СУАМ. Данная инструкция реализуется путем формирования на входе OE преобразователя сигнала «0», как логической функции И-НЕ сигналов $\overline{ME}$ и $\overline{PE}$, равных «1» одновременно. При TST = «0» выполняется следующая по порядку микрокоманда. Данная инструкция может быть использована, например, при вызове подпрограмм обслуживания прерывания или прямого доступа в память.

Инструкция JRP. При TST = «1» выполняется переход на адрес M, источником которого является РМК. Этот адрес устанавливается на входе D СУАМ. При TST = «0» выполняется переход на адрес P, источником которого является внутренний регистр РА СУАМ. Адрес P должен быть предварительно загружен в РА. В отличие от аналогичной инструкции JSRP операция со стеком отсутствует.

Инструкция RFCT. При TST = «0», что интерпретируется как условие «содержимое счетчика не равно 0», выполняется цикл. Это осуществляется переходом к адресу возврата M, который извлекается из верхней ячейки стека (из ячейки СТ0). Адрес M должен быть предварительно запомнен в стеке. Одновременно с этим формируется команда счет для счетчика циклов. При TST = «1», что интерпретируется как условие «содержимое счетчика равно 0», выполняется переход к следующей микрокоманде; при этом адрес возврата (содержимое СТ0) выталкивается из стека. Данная инструкция предназначена совместно с инструкцией JSRP для эффективной организации микропрограммных циклов.

В общем случае адрес M, записанный предварительно в стек, может являться не адресом возврата к началу цикла, а адресом какой-либо произвольной ветви в микропрограммной памяти. Тогда рассматриваемая инструкция может интерпретироваться как «условный переход на адрес из стека по содержимому счетчика».

Инструкция RPCT. Данная инструкция аналогична инструкции RFCT за исключением того, что адрес ветвления извлекается из РМК, а не из стека; содержимое стека не меняется. При TST = «0», что интерпретируется как условие «содержимое счетчика не равно 0», осуществляется переход к адресу возврата, который извлекается из РМК и устанавливается на входе D СУАМ. Одновременно с этим формируется команда «Счет» для счетчика циклов. При TST = «1», что интерпретируется как условие «содержимое счетчика равно 0», выполняется переход к следующей микрокоманде.

В целом инструкция может быть интерпретирована как расширение стека на одно слово в том смысле, что при использовании данной инструкции цикл со счетчиком еще может быть выполнен, когда глубина вложенности достигла пяти.

Инструкция CRTN. При TST = «1» выполняется переход по адресу $N + 1$, извлекаемому из верхней ячейки стека

(адрес $N + 1$ должен быть предварительно запомнен в стеке). После этого содержимое верхней ячейки выталкивается из стека (поскольку после перехода адрес возврата из подпрограммы уже не нужен). При TST = «0» выполняется следующая по порядку микрокоманда подпрограммы.

Инструкция CJPP. При TST = «1» выполняется операция выталкивания из стека, однако переход выполняется к адресу, извлекаемому из РМК на входы D СУАМ. При TST = «0» выполняется следующая по порядку микрокоманда.

Инструкция может быть использована для выхода из тела цикла и подпрограммы до их окончания. При выходе из тела цикла потребность в сохранении адреса возврата исчезает, поэтому он может быть вытолкнут из стека. Инструкция позволяет на аппаратно-программном уровне компактно реализовать таблицы переходов. Для этой цели необходимо в теле цикла последовательно расположить несколько инструкций CJPP (см. табл. 7.35).

Инструкция LDCT. Инструкция безусловная. Формируется команда «Загрузка» для счетчика и выполняется переход к следующей по порядку микрокоманде.

Инструкция LOOP. При TST = «0» выполняется переход по адресу M, извлекаемому из верхней ячейки стека. Адрес M должен быть предварительно запомнен в стеке. При TST = «1» выполняется следующая по порядку микрокоманда, одновременно с этим выталкивается содержимое верхней ячейки стека.

В общем случае адрес M может быть адресом какой-либо произвольной ветви микропрограммной памяти, тогда рассматриваемая инструкция становится инструкцией условного перехода к адресу, извлекаемому из верхней ячейки стека.

Инструкция CONT. Безусловная инструкция, вызывающая увеличение на единицу содержимого счетчика микрокоманд СМК СУАМ и использование этого содержимого в качестве адреса следующей микрокоманды.

Инструкция JP. Выполняется безусловный переход к адресу, извлекаемому из РМК. Этот адрес (M) устанавливается на входе D СУАМ.

7.6. ПАРАЛЛЕЛЬНЫЙ РЕГИСТР К1804ИР1

Регистр К1804ИР1 предназначен для хранения информации и может быть использован в самых разнообразных цифровых устройствах общего и специального назначения. Условное графическое обозначение регистра приведено на рис. 7.36, назначение выводов описано в табл. 7.36. Функциональная схема регистра представлена на рис. 7.37.

Регистр состоит из четырех триггеров D-типа. Запись информации со входов $D3 \ldots D0$ происходит по положительному фронту тактового импульса T. На шины $Q3 \ldots Q0$ выводится информация непосредственно с прямых выходов триггеров. Выходные шины

$Y3 \ldots Y0$ снимаются с трехстабильных выходов регистра, управляемых сигналом $\overline{OE}$. При $\overline{OE}$ = «0» шины $Y3 \ldots Y0$ воспроизводят информацию, содержащуюся в регистре, т. е. то же, что и шины $Q3 \ldots Q0$. При $\overline{OE}$ = «1» выходы $Y3 \ldots Y0$ отключаются

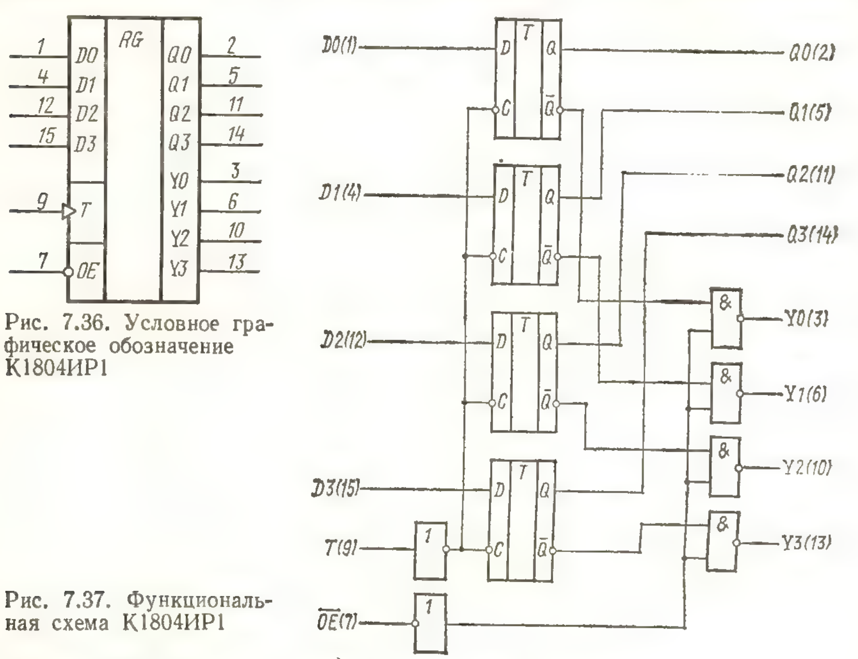


Рис. 7.36. Условное графическое обозначение К1804ИР1

Рис. 7.37. Функциональная схема К1804ИР1

Таблица 7.36. **Назначение выводов микросхемы К1804ИР1**

Номер вывода	Назначение	Обозначение	Разряд	Тип
1	Данные	$D0$	0	Вход
2		$Q0$	0	Выход
3	Управляемый выход	$Y0$	0	
4	Данные	$D1$	1	Вход
5		$Q1$	1	Выход
6	Управляемый выход	$Y1$	1	
7	Разрешение выходов Y	OE	—	Вход
8	Общий	GND	—	—
9	Тактовый	T	—	Вход
10	Управляемый выход	$Y2$	2	Выход
11	Данные	$Q2$	2	
12		$D2$	2	Вход
13	Управляемый выход	$Y3$	3	Выход
14	Данные	$Q3$	3	
15		$D3$	3	Вход
16	Питание	U_{CC}	—	—

Примечание. Выходы $Y3 \ldots Y0$ трехстабильные.

(находятся в третьем состоянии). Особенностями регистра являются дублированная система прямых выходов и относительно малая задержка распространения сигнала от тактового входа к выходам.

Статические и динамические параметры К1804ИР1 даны в табл. 7.37 и 7.38 соответственно. Временна́я диаграмма работы представлена на рис. 7.38.

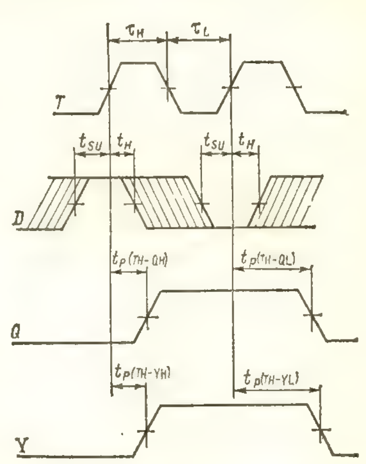


Рис. 7.38. Временна́я диаграмма работы К1804ИР1

Параллельный регистр К1804ИР1 может использоваться в разработках микропроцессорных устройств в качестве адресного регистра, регистра состояния, регистра команд или каких-либо регистров данных или микрокоманд. Кроме того, использование прямых и управляемых выходов регистра дает возможность построения на базе К1804ИР1 некоторых специализированных устройств, например, схем интерфейса и преобразования кодов.

На рис. 7.39 показано применение двух микросхем К1804ИР1 для организации двунаправленного интерфейса между двумя шинами. Левый регистр принимает данные с шины *А* и может передавать их на шину *Б*. Правый

Таблица 7.37. **Статические параметры микросхемы К1804ИР1**

Параметр, единица измерения	Обозначение	Значения параметров		Режим измерения
		мин.	макс.	
Входной ток «0», мА	I_{IL}	—	—2,0	$U_{CC} = 5{,}25$ В $U_{IL} = 0{,}5$ В
Входной ток «1», мкА	I_{IH}	—	50	$U_{CC} = 5{,}25$ В $U_{IH} = 2{,}7$ В
Выходное напряжение «0», В	U_{OL}	—	0,5	$U_{CC} = 4{,}75$ В $U_{TL} = 0{,}8$ В $U_{IH} = 4{,}75$ В $U_{IL} = 0$ В $I_{OL} = 20$ мА

Параметр, единица измерения	Обозначение	Значения параметров		Режим измерения
		мин.	макс.	
Выходное напряжение «1», В	U_{OH}	2,4	—	$U_{CC} = 4{,}75$ В $U_{TH} = 2{,}0$ В $U_{IH} = 4{,}75$ В $U_{IL} = 0$ В $I_{OH} = -1$ мА (для $Q3 \ldots Q0$) $I_{OH} = -2$ мА (для $Y3 \ldots Y0$)
Прямое падение напряжения на антизвонном диоде, В	U_{CDI}	—	1,2	$U_{CC} = 4{,}75$ В $I_I = -18$ мА
Выходной ток в состоянии «выключено» при напряжении «0» на выходе, мкА	I_{OZL}	—	—50	$U_{CC} = 5{,}25$ В $U_{OL} = 0{,}4$ В
Выходной ток в состоянии «выключено» при напряжении «1» на выходе, мкА	I_{OZH}	—	50	$U_{CC} = 5{,}25$ В $U_{OH} = 2{,}4$ В
Ток короткого замыкания на выходе, мА	I_{OS}	—40	—100	$U_{CC} = 5{,}25$ В $U_{OL} = 0$ В
Ток потребления, мА	I_{CC}	—	130	$U_{CC} = 5{,}25$ В

регистр принимает данные от шины *B* и может передавать их на шину *A*. Управление выходами *Y* используется для установки выходов любого из регистров в состояние высокого сопротивления. Содержимое каждого регистра доступно для непрерывного использования на выходах *N* (левого регистра) и *M* (правого).

На рис. 7.40 показано применение двух микросхем К1804ИР1 для организации восьмиразрядного преобразователя последовательного кода в параллельный. Вход *D*0 левого регистра служит входом последовательных данных. Выходы *Q* соединены со входами *D* последующих разрядов. Параллельный код снимается с трехстабильных выходов *Y* на шину *W*. Непосредственный доступ к содержимому регистров возможен через шину *X*.

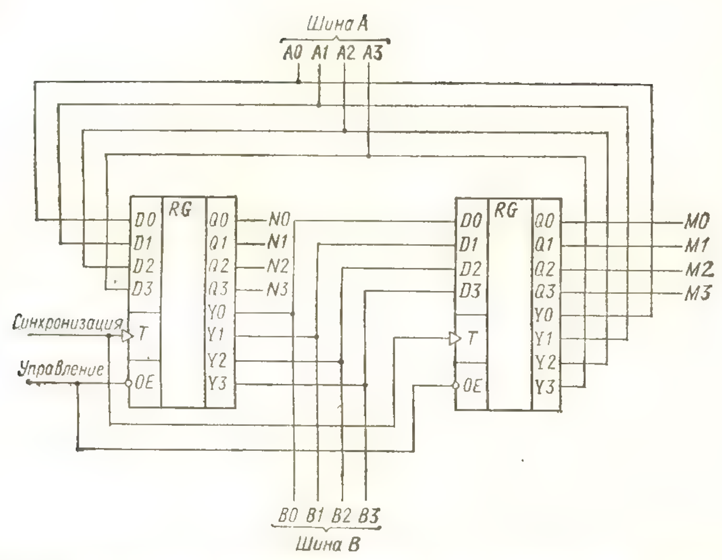


Рис. 7.39. Организация интерфейса между двумя шинами на К1804ИР1

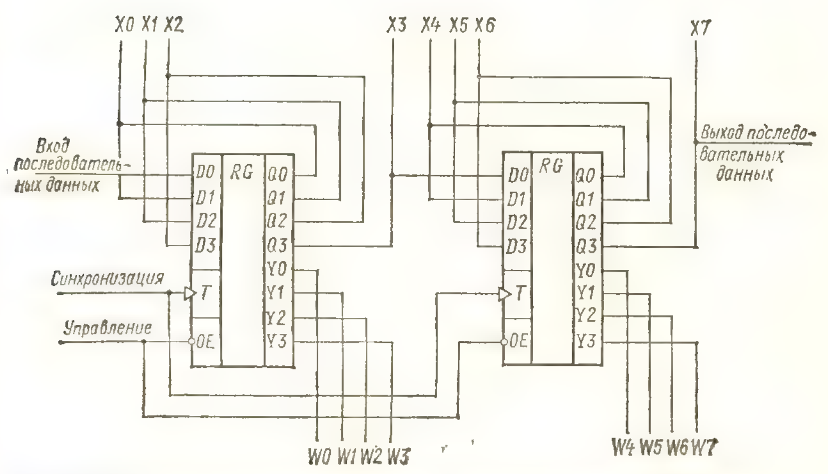


Рис. 7.40. Последовательно-параллельный преобразователь на К1804ИР1

Таблица 7.38. **Динамические параметры микросхемы К1804ИР1**

Параметр	Значение, нс		Температура, °С
	мин.	макс.	
Длительность сигнала низкого уровня τ_L на входе T	9 15	— —	25 —10 ... 70
Длительность сигнала высокого уровня τ_H на входе T	7 10	— —	25 —10 ... 70
Время задержки распространения сигнала $t_р$ от входа T до выходов Y	3 5	21 22	25 —10 ... 70
Время установления сигнала t_{SU} на входах D относительно положительного перепада сигнала на входе T	5 7	— —	25 —10 ... 70
Время сохранения сигнала t_H на входах D относительно положительного перепада сигнала на входе T	3 5	— —	25 —10 ... 70
Время задержки распространения сигнала при переходе из состояния «выключено» в состояние «1» t_{ZH} от входа OE до выходов Y	— —	27 30	25 —10 ... 70
Время задержки распространения сигнала при переходе из состояния «выключено» в состояние «0», t_{ZL} от входа OE до выходов Y	— —	26 30	25 —10 ... 70
Время задержки распространения сигнала при переходе из состояния «1» в состояние «выключено», t_{HZ} от входа OE до выходов Y	— —	6 10	25 —10 ... 70
Время задержки распространения при переходе из состояния «0» в состояние «выключено» t_{LZ} от входа OE до выходов Y	— —	11 15	25 —10 ... 70

Примечание. Временные параметры измерялись при $U_{CC} = 5 \pm 0{,}25$ В и $C_L = 50$ пФ.

7.7. ОСОБЕННОСТИ ПРИМЕНЕНИЯ МПК БИС СЕРИИ К1804

МПК БИС серии К1804 может быть использован для разработки широкого спектра устройств вычислительной техники, начиная от простых контроллеров и кончая мультипроцессорными системами. Рассмотрим некоторые особенности архитектуры процессоров на базе комплекта К1804 с расчетом временных параметров, а также конкретный пример построения аппаратуры на базе БИС серии К1804.

Типовая схема устройства обработки информации (центрального процессора микроЭВМ), построенного на основе микросхем комплекта, представлена на рис. 7.41. В устройстве можно выделить две части: устройство микропрограммного управления и устройство обработки данных.

Устройство микропрограммного управления включает *регистр команд*, *дешифратор команд*, *схему управления следующим*

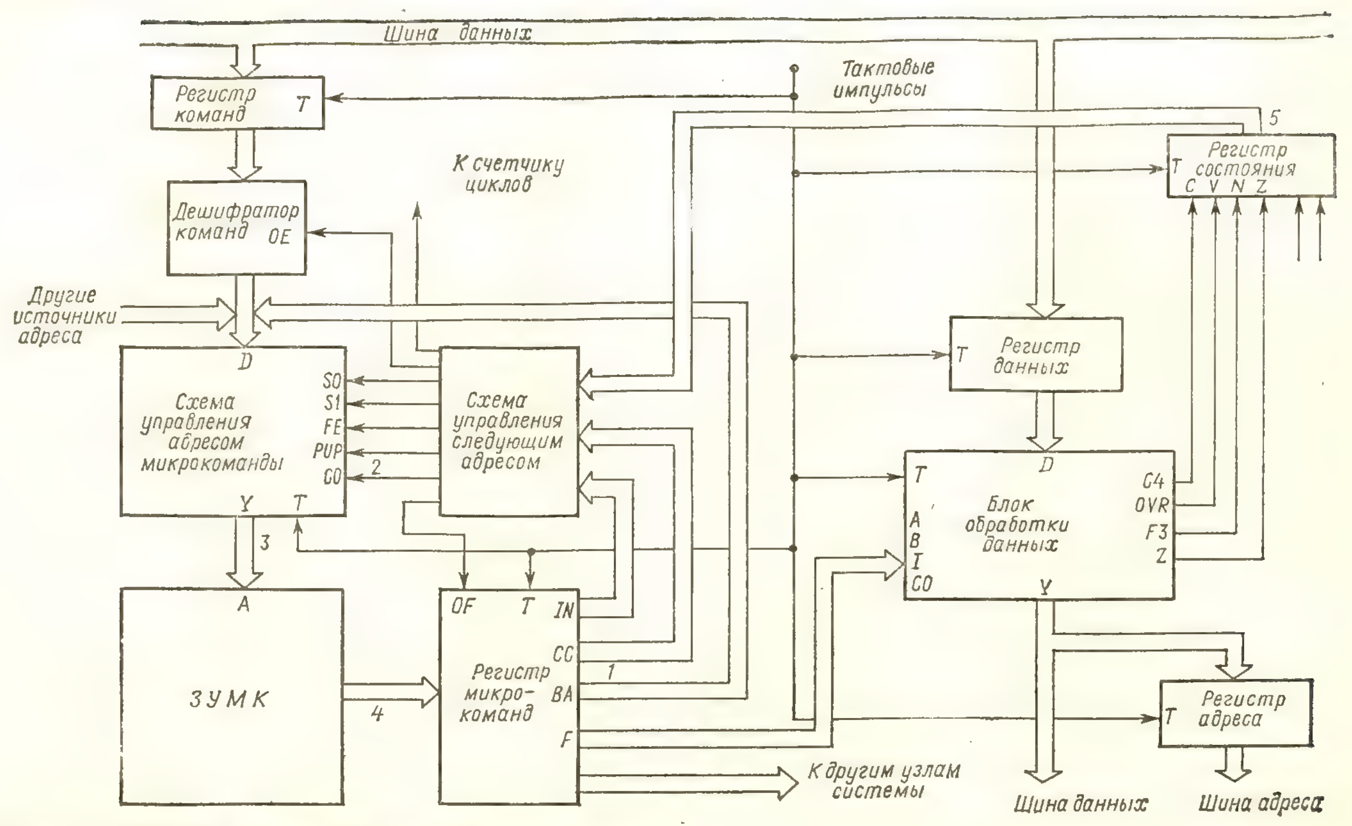


Рис. 7.41. Типовая схема устройства обработки информации

адресом, схему управления адресом микрокоманды, запоминающее устройство микрокоманд *ЗУМК*, *регистр микрокоманд*. Регистр команд может быть построен на микросхемах К1804ИР1. Разрядность его соответствует разрядности команды, выбираемой из основной памяти по шине данных. По синхросигналу команда записывается в регистр команд и поступает на входы дешифратора команд, который преобразует команду, записанную в регистре команд, в адрес микрокоманды. Разрядность выходов дешифратора команд равна разрядности адреса микрокоманды и определяет также объем *ЗУМК*. Адрес с выхода дешифратора команд подается на входы *D* схемы управления адресом микрокоманды. В качестве дешифратора команд может быть применено ППЗУ, имеющее трехстабильные выходы, поскольку выход дешифратора команд подключается к общей шине. Управление состоянием трехстабильных выходов дешифратора команд осуществляется схемой управления следующим адресом. Схема управления адресом микрокоманды может быть построена из секций управления адресом микрокоманды СУАМ1 (К1804ВУ1) или СУАМ2 (К1804ВУ2). Разрядность входа *D*, выхода *Y* и всех внутренних регистров должна быть равной разрядности адреса микрокоманды (т. е. разрядности входа *A ЗУМК*). Секции СУАМ, из которых строится схема управления адресом микрокоманд, имеют общие линии управления и синхронизации; перенос между секциями — последовательный. Эта схема вырабатывает адрес, который подается на вход *A ЗУМК*. *ЗУМК* содержит слова микрокоманды, длина которых (число разрядов) определяется требованиями системы. Объем *ЗУМК* определяется разрядностью адреса. При 12-разрядном адресе можно адресоваться к *ЗУМК* объемом 4096 слов.

Регистр микрокоманд может быть построен на микросхемах К1804ИР1. При этом, если часть сигналов микрокоманды с него поступает на общую шину (например, сигналы, идущие на входы *D* схемы управления адресом микрокоманды), используют трехстабильные выходы *Y* К1804ИР1. Разрядность регистра микрокоманд соответствует радрядности микрокоманды, выбираемой из *ЗУМК*. Каждая микрокоманда содержит биты для управления всеми узлами устройства обработки данных. Любая элементарная операция этого устройства будет исполнена, когда сигналы микрокоманды поступят на узлы устройства обработки данных и будет подан синхросигнал.

Поскольку команда, поступающая по шине данных, выполняется как последовательность микрокоманд, то любая микрокоманда должна содержать биты не только для управления устройством обработки данных, но и для выборки из *ЗУМК* следующей микрокоманды. Кроме того, микрокоманда осуществляет управление и другими узлами системы. Таким образом, микрокоманда, находящаяся в регистре микрокоманд, состоит из нескольких полей. Поле *IN* (см. рис. 7.41) содержит сигналы для схемы управления следующим адресом (они указывают, где содержится следующий

адрес: в регистре *РА*, в *СМК*, в стеке или на входах *D* СУАМ, а также будет ли выполняться запись адреса в стек или извлечение из него. Поле *СС* содержит сигналы, определяющие условия для выполнения сигналов поля *IN*. Совместно с сигналами регистра состояния сигналы поля *СС* могут модифицировать сигналы поля *IN* (модификация сигналов осуществляется в схеме управления следующим адресом). Поле *ВА* содержит сигналы, указывающие адрес перехода или адрес подпрограммы. Поле *F* содержит сигналы, управляющие блоком обработки данных.

Регистр микрокоманд выполняет функцию конвейерного регистра. Пока исполняется одна микрокоманда, следующая (на основании сигналов полей *СС*, *IN* и *ВА*) выбирается из *ЗУМК* и будет записана в него с приходом синхросигнала. Таким образом, два действия выполняются параллельно. За счет этого тактовая частота увеличивается по сравнению с последовательным выполнением этих действий.

Схема управления следующим адресом служит для подачи сигналов на схему управления адресом микрокоманды, дешифратор команд, счетчик циклов, регистр микрокоманд. Эти сигналы формируются на основе сигналов регистра микрокоманд и регистра состояния. Схема управления следующим адресом является комбинационной схемой типа ПЗУ или ПЛМ и может быть построена на микросхеме К1804ВУ3 (УСА).

Устройство обработки данных включает: *регистр данных*, *блок обработки данных*, *регистр адреса*, *регистр состояния*. Регистр данных служит для хранения данных, поступивших из основной памяти по шине данных. Запись в регистр данных осуществляется по синхросигналу, и его разрядность соответствует разрядности обрабатываемых операндов и блока обработки данных. Регистр данных может быть построен на микросхемах К1804ИР1. Его содержимое может либо записываться во внутренние регистры блока обработки данных, либо непосредственно использоваться в качестве операндов для АЛУ блока обработки данных.

Блок обработки данных строится на базе микропроцессорных секций К1804ВС1 с использованием схем ускоренного переноса К1804ВР1. Работа этих схем, правила их соединения и наращивания подробно описаны в разд. 7.3. Управляется блок обработки данных полем *F* микрокоманды, которое включает адреса *А* и *В* блока внутренней памяти микропроцессорной секции, входной сигнал *С*0 и девятиразрядное поле *I*, управляющее операциями микропроцессора. Данные с этого блока поступают на шину данных и на регистр адреса.

Вырабатываемые микропроцессором признаки нуля результата, переноса, переполнения и знакового разряда выдаются на регистр состояния. Регистр адреса служит для хранения адресов в случае их выработки блоком обработки данных, и его разрядность соответствует разрядности адресного поля основной памяти. Регистр может быть построен на микросхемах К1804ИР1 с использо-

ванием трехстабильных выходов, так как его выходы подсоединяются к общей шине адреса. Регистр состояния служит для хранения слова состояния системы. Он также может быть построен на микросхемах К1804ИР1. Запись в него происходит по синхросигналам после каждой операции блока обработки данных. Содержимое регистра состояния поступает на входы схемы управления следующим адресом.

В целом рассмотренное устройство обработки информации функционирует следующим образом. Регистр микрокоманд (конвейерный регистр) содержит текущую (исполняемую) микрокоманду *1* (см. рис. 7.41). Часть микрокоманды, управляющая обработкой данных, передается на блок обработки данных и к другим исполнительным узлам системы. Другая часть микрокоманды преобразуется и возвращается к схеме управления адресом микрокоманды *2* для того, чтобы определить адрес следующей, подлежащей выполнению микрокоманды. Этот адрес *3* пересылается в *ЗУМК*, и следующая микрокоманда *4* выбирается из *ЗУМК*. Цепочка действий *1—2—3—4* выполняется в течение одного синхропериода. В течение того же самого времени блок обработки данных должен произвести необходимые операции с данными и выполнить цепочку действий *1—5*. Микросхемы микропроцессорного комплекта К1804 обеспечивают примерно одинаковую длительность времен выполнения обеих цепочек.

После окончания выполнения необходимого количества микрокоманд слово, находящееся в одном из регистров РЗУ микропроцессора К1804ВС1 блока обработки данных, увеличенное на единицу, передается в регистр адреса. По этому адресу из памяти данные загружаются в регистр команд. Содержимое регистра команд с помощью дешифратора команд и схемы управления адресом микрокоманды генерирует адрес первой микрокоманды, которая выбирается из ЗУМК и поступает на регистр микрокоманд, после чего микрокоманда выполняется. В соответствии с требованием команды может быть выполнено несколько микрокоманд, осуществляющих выбор данных из памяти, операции АЛУ, проверку на переполнение и т. п. При этом все действия соответствуют выполнению цепочек *1—2—3—4* и *1—5*. Затем происходит обращение к циклу выборки команды. В этот момент может произойти переход к другим источникам адреса микрокоманд, например, в результате прерывания, когда вместо выполнения новой команды устройство может начать выполнение процедуры, обслуживающей прерывание.

Рассмотрим методику расчета временных параметров устройства обработки данных. Методику расчета поясняет схема (рис. 7.42), содержащая: четыре МПС К1804ВС1 (показаны старшая *СТМПС* и младшая схемы *МЛМПС*) и *СУП* К1804ВР1 (эти пять схем составляют блок обработки данных), регистр данных *PD*, регистр микрокоманд *РМК*, регистр результата *РР*, регистр состояния *PC*, мультиплексоры *М*, управляющие сдвигами и за-

мыкающие сдвиговые выводы крайних МПС. Все МПС и регистры синхронизируются от одного источника тактовых сигналов Т. Регистры данных, микрокоманды, результата и состояния построены на микросхемах К1804РИ1. Временные параметры, принятые в расчетах, даны для температуры 25 °С.

Все операции, выполняемые схемой (см. рис. 7.42), с точки зрения особенностей расчета временных параметров, можно классифицировать на следующие группы: логические и арифметические; со сдвигом и без сдвига.

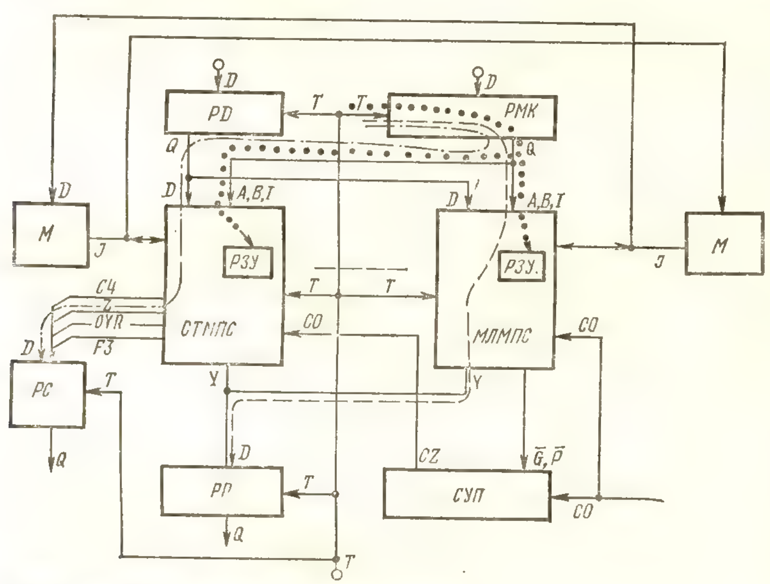


Рис. 7.42. Логические операции без сдвигов

В рассматриваемой схеме источниками информации являются *PD*, *PMK*, *РЗУ*; приемниками информации являются *PP*, *PC*, *РЗУ*. Полагая, что задержки от источников информации определяются величиной задержки микросхемы К1804ИР1, для каждой группы операций рассмотрим три пути прохождения информации, отличающиеся местоположением приемников информации.

Путь 1 заканчивается фиксацией результата в одном из регистров *РЗУ*, путь 2 — фиксацией результата в *PP*, путь 3 — фиксацией признаков в *PC*.

Рассмотрим логические операции без сдвигов. На рис. 7.42 показаны пути прохождения информации, на рис. 7.43 приведена временнáя диаграмма, а в табл. 7.39 — расчет длительности путей прохождения информации.

По положительному фронту тактового сигнала считывается информация в регистрах-источниках — *PD* и *PMK*. С задержкой

($T \to Q$), равной в данном случае 21 нс, эта информация появляется на входах A, B, I, D секций.

Первый путь Чтение—Модификация—Запись (Чт—Мод—Зп) включает внутренние переходные процессы в секциях, состоящие в следующем: выборка информации из *РЗУ* по адресам A и B и прохождение ее на один из входов *АЛУ*, настройка *АЛУ* на требуемую операцию, поступление с шины D информации на другой

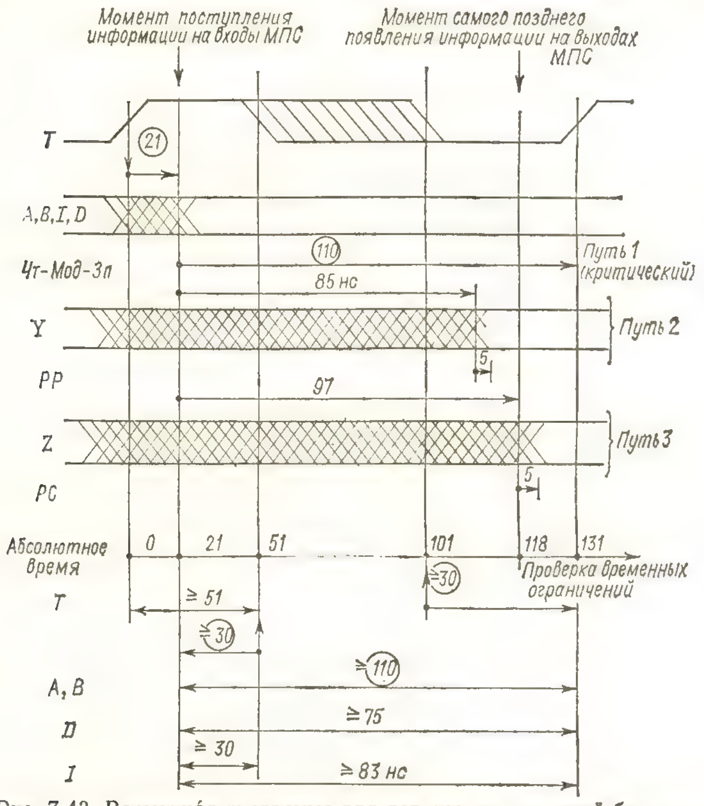


Рис. 7.43. Временна́я диаграмма для логических операций без сдвигов

вход *АЛУ*, выполнение операции в *АЛУ*, прохождение результата с выхода *АЛУ* на вход *РЗУ*.

Указанные процессы протекают при «1» на тактовом входе. При переходе тактового сигнала в «0» разрешается запись и происходит фиксация результата операции в одном из регистров *РЗУ*. Длительность цикла Чт—Мод—Зп равна 110 нс, т. е. общая длительность первого пути равна 131 (21 + 110) нс.

На втором пути учитывается прохождение информации с выхода *АЛУ* на выходную шину Y, причем длина этого пути оцени-

Таблица 7.39. **Время выполнения логических операций без сдвига**

Узел	Путь в узле	Время, нс		
		Путь 1	Путь 2	Путь 3
Рг источника	$T \to Q$	21	21	21
МПС	Чт—Мод—Зп	110	—	—
МПС	$A, B \to Y$	—	85	—
МПС	$A, B \to Z$	—	—	97
Рг приемника	Установка D	—	5	5
Итого, задержка в МПС	—	110	85	97
Итого	—	131	111	123

вается суммарной задержкой распространения информации с адресных входов A и B (как наибольшая) до выхода Y, которая равна 85 нс. Заканчивается второй путь установкой информации на входе D регистра результата. Для надежной фиксации этой информации по очередному положительному фронту на тактовом входе время установки должно быть не менее 5 нс. Таким образом, суммарная длина второго пути равна 111 (21 + 85 + 5) нс.

Третий путь связан с выработкой и фиксацией признаков результата, которые появляются при логических операциях, например, признак нуля. Выработка этого признака происходит только в старшей секции и занимает время от момента поступления информации на входы МПС (через 21 нс после первого положительного фронта тактового импульса) до появления информации на выходах признаков. Дольше всего формируется признак Z, за 97 нс. Завершается третий путь, как и второй, установкой информации на входе регистра приемника (PC) в течение 5 нс. Общая длительность третьего пути равна 123 (21 + 97 + 5) нс.

В нижней части рис. 7.43 приведены для контроля ограничения на времена установки сигналов на входах МПС. Эти ограничения, в частности, определяют, что длительность положительного импульса тактового периода не может быть меньше 51 нс, так как время от начала появления информации на адресных входах A, B и входах микрокоманды I до отрицательного фронта тактового сигнала T не менее 30 нс. Длительность отрицательного импульса тактового периода не должна быть меньше 30 нс. Таким образом, отрицательный фронт тактового сигнала может быть расположен в интервале от 51 до 101 нс от начала такта.

Рассмотрим арифметические операции без сдвигов. Расчет длительности путей прохождения информации для арифметических операций без сдвига представлен в табл. 7.40. Пути прохождения

информации показаны на рис. 7.44, временная диаграмма — на рис. 7.45. В арифметических операциях при формировании переноса участвует схема К1804ВР1.

Первый путь включает выработку сигналов на выходах *G*, *P* за 70 нс, задержку сигнала в *СУП* — 15 нс, после чего информация поступает на вход *С*0 старшей секции. Для надежной записи

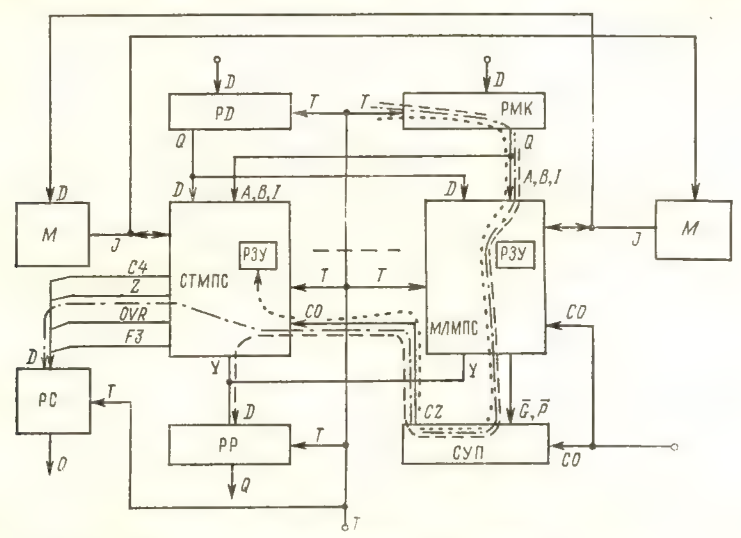


Рис. 7.44. Арифметические операции без сдвигов

результата операции в *РЗУ* время установки сигнала на этом входе должно быть не менее 60 нс до второго перехода тактового сигнала с «0» в «1». В рассматриваемом случае этот путь, длиной в 166 нс,

Таблица 7.40. **Время выполнения арифметических операций без сдвига**

Узел	Путь в узле	Время, нс		
		Путь 1	Путь 2	Путь 3
Рг источника	$T \to Q$	21	21	21
МПС	$A,\ B \to G,\ P$	70	30	70
СУП	$G,\ P \to C,\ Z$	15	15	15
СТМПС	Установка *С*0	60	—	—
СТМПС	$C0 \to Y$	—	35	—
СТМПС	$C0 \to Z$	—	—	52
Рг приемника	Установка *D*	—	5	5
Итого, задержка в МПС	—	130	105	122
Итого	—	166	146	163

является критическим. На втором пути учитывается время задержки сигнала от входа $C0$ старшей секции до выхода Y, равное 35 нс. На третьем пути учитывается наибольшее время задержки от входа $C0$ старшей секции до выходов признаков. Дольше всех формируется признак Z, за 52 нс.

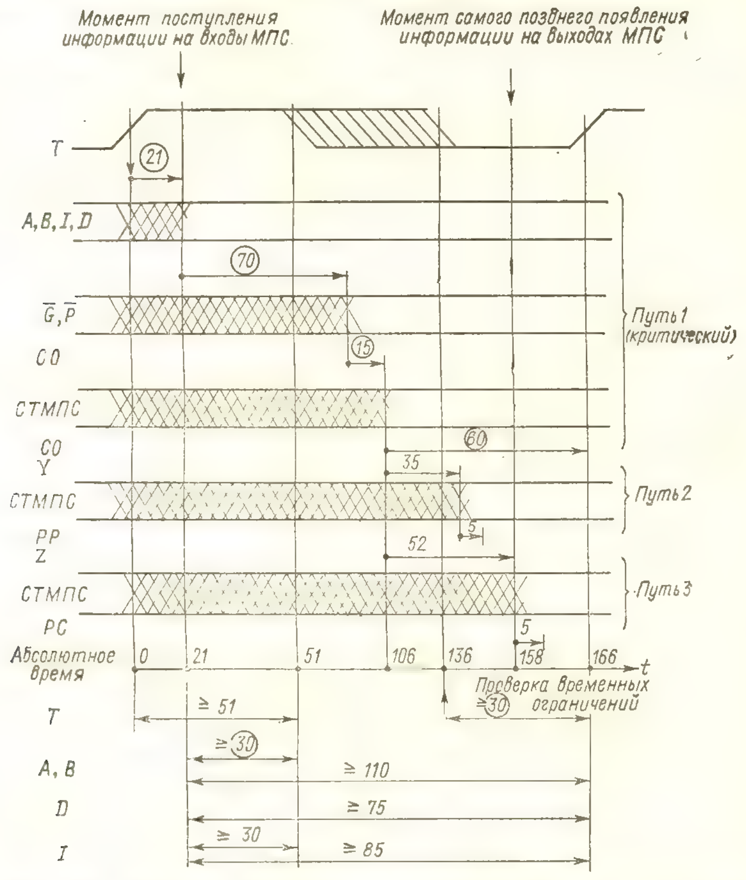


Рис. 7.45. Временнáя диаграмма для арифметических операций без сдвигов

Операции со сдвигом. Во всех операциях со сдвигом фиксация сдвинутой информации возможна только на внутренних приемниках информации — регистрах *РЗУ* или *PQ*; на шину *Y* с выхода АЛУ выдается несдвинутая информация, поэтому в логических и арифметических операциях изменяются только первые пути. Расчет длительности путей прохождения информации для логических операций со сдвигом поясняется в табл. 7.41 и рис. 7.46.

Примем для определенности, что выполняется сдвиг вправо (результаты расчета задержек не зависят от направления сдвига).

Первый этап, как и в предыдущих случаях, — считывание информации с регистров источников длительностью 21 нс. Далее — чтение операндов из РЗУ, выполнение операции в АЛУ и появление на одном из правых сдвиговых выводов младшей секции младшего разряда результата. Длительность этого этапа 100 нс. Далее с задержкой на мультиплексоре в 20 нс информация поступает

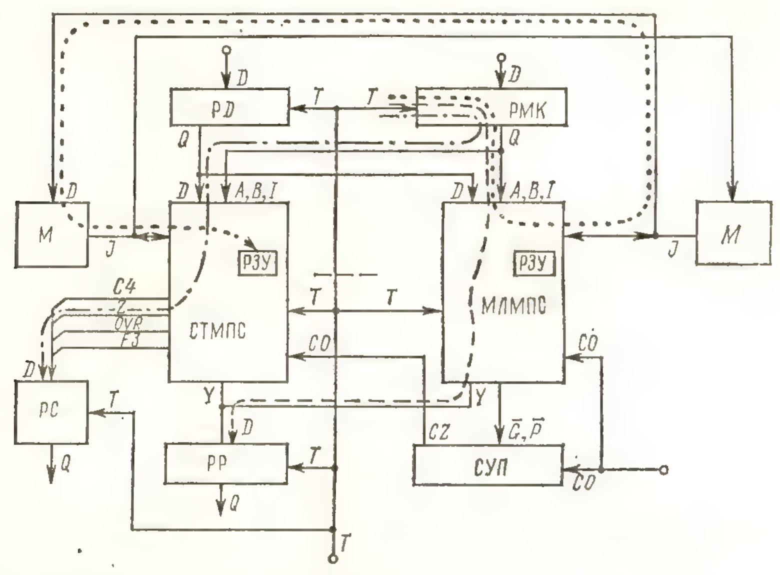


Рис. 7.46. Логические операции со сдвигом

на левый сдвиговый вход старшей секции. Для уверенной записи сдвинутого результата во внутренний регистр время установки

Таблица 7.41. **Время выполнения логических операций со сдвигом**

Узел	Путь в узле	Время, нс		
		Путь 1	Путь 2	Путь 3
Рг источника	$T \to Q$	21	21	21
МПС	$A, B \to PR0, PR3$	100	—	—
М	$D \to J$	20	—	—
МПС	Установка $RP0, RP3$	25	—	—
МПС	$A, B \to Y$	—	85	—
МПС	$A, B \to Z$	—	—	97
Рг приемника	Установка D	—	5	5
Итого, задержка в МПС	—	125	85	97
Итого	—	166	111	123

информации на этом входе должно быть не менее 25 нс. Общая длительность пути 166 нс.

Пути с фиксацией информации на внешних регистрах аналогичны соответствующим путям для логических операций без сдвигов.

Расчет длительности путей прохождения информации для арифметических операций с кодами со сдвигом поясняется табл. 7.42 и рис. 7.47. Расчет длительности путей прохождения информации для арифметических операций со сдвигом в дополнительном коде поясняется табл. 7.43 и рис. 7.48.

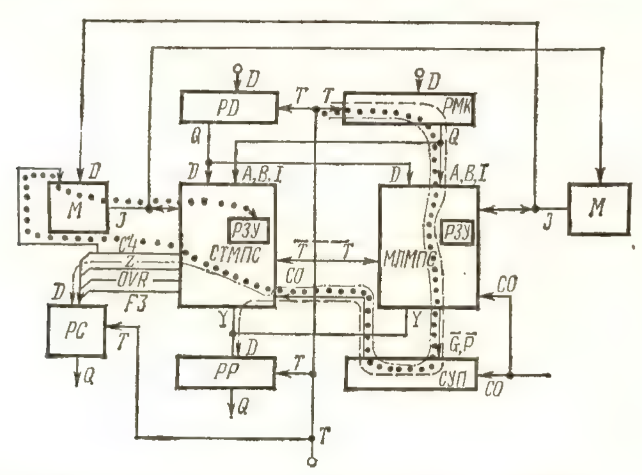


Рис. 7.47. Арифметические операции с кодами со сдвигом вправо

При расчете длительности пути с фиксацией сдвинутой информации на внутреннем регистре, кроме задержек на *СУП* и мультиплексоре, учитывается задержка на логической схеме Исключающее ИЛИ, формирующей из сигналов признаков содержимое старшего (вдвигаемого) разряда результата.

Теперь рассмотрим расчет временных параметров устройства микропрограммного управления. Пусть устройство микропрограммного управления имеет структуру, представленную на рис. 7.35. Расчет длительности путей прохождения информации в цикле микропрограммного управления поясняется рис. 7.49 и табл. 7.44—7.47. Значения временны́х параметров в табл. 7.44—7.47 даны для температуры 25 °С. Предполагается, что в схеме используются следующие узлы: схема управления следующим адресом *УСА* — К1804ВУ3; схема управления адресом микрокоманды *СУАМ* — К1804ВУ1; запоминающее устройство микрокоманд *ЗУМК* КР556РТ5; регистр микрокоманд К1804ИР1; мультиплексор кода условий (*М*) — 133КП5; схема Исключающее ИЛИ — 133ЛП5; преобразователь адреса имеет управляемый выход с тремя состояниями, аналогичный микросхеме К1804ИР1. Число корпусов микросхем, используемых для построения *СУАМ*, *ЗУМК*

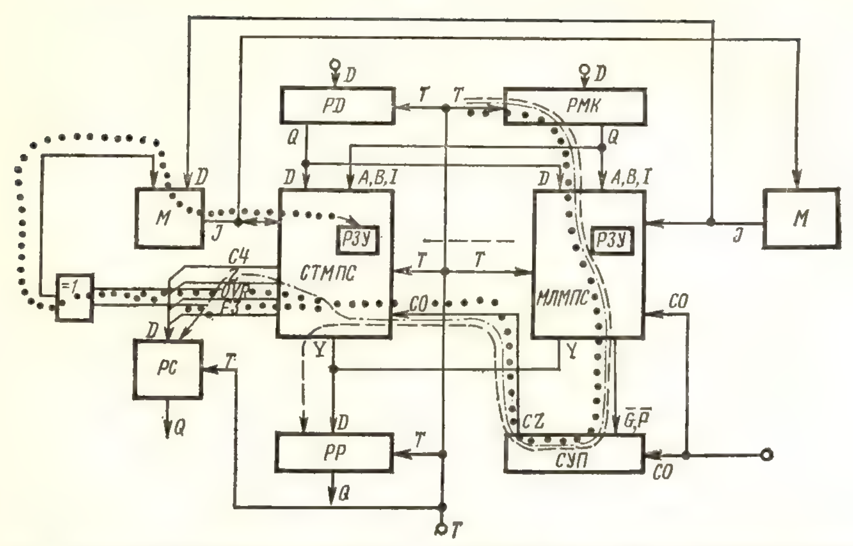


Рис. 7.48. Арифметические операции в дополнительном коде со сдвигом вправо

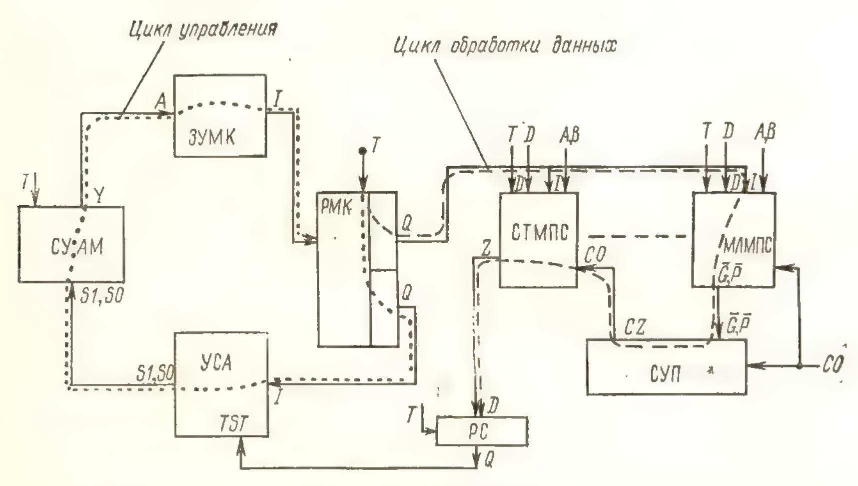


Рис. 7.49. Циклы управления и обработки данных

Таблица 7.42. **Время выполнения арифметических операций с кодами со сдвигом**

Узел	Путь в узле	Время, нс		
		Путь 1	Путь 2	Путь 3
Рг источника	$T \to Q$	21	21	21
МПС	$A, B \to G, P$	70	70	70
СУП	$G, P \to C, Z$	15	15	15
МПС	$C0 \to C4$	25	—	—
М	$D \to J$	20	—	—
МПС	$C0 \to Y$	—	—	35
МПС	$C0 \to Z$	—	52	—
Рг приемника	Установка D	—	5	5
МПС	Установка $PR0$, $PR3$	25	—	—
Итого, задержка в МПС	—	120	122	105
Итого	—	176	163	146

и регистра микрокоманд будет определяться числом микрокоманд и их разрядностью.

Длительности путей прохождения информации будут зависеть от включения в цикл микрокоманды тех или иных узлов схемы и их быстродействия. Определение этих путей зависит в основном от типа инструкций *УСА* (К1804ВУ3). Ниже приводятся примеры

Таблица 7.43. **Время выполнения арифметических операций в дополнительном коде со сдвигом**

Узел	Путь в узле	Время, нс		
		Путь 1	Путь 2	Путь 3
Рг источника	$T \to Q$	21	21	21
МПС	$A, B \to G, P$	70	70	70
СУП	$G, P \to C, Z$	15	15	15
МПС	$C0 \to F3, OVR$	35	—	—
Логическая схема	Вход → Выход	23	—	—
М	$D \to J$	20	—	—
МПС	Установка $PR0$, $PR3$	25	—	—
МПС	$C0 \to Y$	—	35	—
МПС	$C0 \to Z$	—	—	52
Рг приемника	Установка D	—	5	5
Итого, задержка в МПС	—	130	105	122
Итого	—	209	146	163

Таблица 7.44. **Время выполнения безусловного перехода по адресу из РМК**

Узел	Путь в узле	Время, нс
РМК	$T \to Q$	21
УСА	$I \to PE$	60
РМК	$OE \to Y$	27
СУАМ	$D \to Y$	20
ЗУМК	$A \to I$	70
РМК	Установка D	5
Итого	—	203

Таблица 7.45. **Время выполнения условного перехода по адресу из регистра адреса**

Узел	Путь в узле	Время, нс
РМК	$T \to Q$	21
М	$I \to Y$	35
Схема «Исключающее ИЛИ»	Вход → Выход	23
УСА	$TTS \to S0,\ S1$	60
СУАМ	$S0,\ S1 \to (PA) \to Y$	20
ЗУМК	$A \to I$	70
РМК	Установка D	5
Итого		234

Таблица 7.46. **Время выполнения условного перехода на адрес вектора**

Узел	Путь в узле	Время, нс
РМК	$T \to Q$	21
М	$I \to Y$	35
Схема «Исключающее ИЛИ»	Вход → Выход	23
УСА	$TST \to ME,\ PE$	60
Схема «И—НЕ»	Вход → Выход	22
Выход преобразователя адреса	$OE \to Y$	27
СУАМ	$D \to Y$	20
ЗУМК	$A \to I$	70
РМК	Установка D	5
Итого		283

Таблица 7.47. **Время выполнения перехода по адресу из счетчика микрокоманд**

Узел	Путь в узле	Время, нс
РМК	$T \to Q$	21
УСА	$I \to S0,\ S1$	60
СУАМ	$S0,\ S1 \to$ (СМК) $\to Y$	20
ЗУМК	$A \to I$	70
РМК	Установка D	5
Итого		176

расчетов длительности прохождения информации для ряда инструкций *УСА*, согласно схеме рис. 7.49. На рис. 7.49 условно показаны пути прохождения информации в соответствии с этими инструкциями.

При расчетах началом цикла считается положительный фронт синхросигнала, по которому микрокоманда записывается в *РМК*. В результате работы устройства микропрограммного управления к концу цикла на входы *РМК* из *ЗУМК* поступает новая микрокоманда.

Безусловный переход на адрес из регистра микрокоманд, инструкция JP (путь 1). По положительному фронту синхросигнала фиксируется информация в *РМК*. С задержкой ($T \to Q$) эта информация появляется на управляющих входах *I УСА*. С задержкой 60 нс *УСА* вырабатывается сигнал $\overline{PE}$, поступающий на вход разрешения выходов *Y РМК* (вход *OE*). С задержкой $OE \to Y$ (27 нс) информация с *РМК* (новый адрес) поступает на входы *D* СУАМ. К этому времени схема *УСА* сформировала необходимые сигналы *S1*, *S0* на входах СУАМ, и с задержкой $D \to Y$ новый адрес поступает с выходов СУАМ на адресные входы *ЗУМК*. С задержкой $A \to I$, равной времени считывания из *ЗУМК*, новая микрокоманда поступает на входы *РМК*. Для надежной фиксации новой микрокоманды в *РМК* требуется время установки (5 нс) на входах *РМК* перед приходом положительного фронта синхросигнала.

Условный переход к подпрограмме на адрес регистра адреса СУАМ, инструкция JSRP (путь 2). При выполнении данной инструкции сигналы от *РМК* управляют наряду с *УСА* логической схемой инверсии кода условия и мультиплексором кода условия (*М*). На информационные входы мультиплексора сигналы поступают с регистра состояния, который синхронизируется одновременно с *РМК*. Полагая, что *РС* собран на микросхемах К1804ИР1, можно считать, что информация на информационных входах присутствует одновременно с управляющими сигналами от *РМК*. Поскольку в мультиплексоре 133КП5 задержка от управляющих входов пре-

вышает задержку от информационных входов, то она и будет определять потери времени на мультиплексоре ($I \rightarrow Y$). С мультиплексора с задержкой на логической схеме 133ЛП5 информация поступает на вход T УСА. Через 60 нс ($TST \rightarrow S0$, $S1$) УСА подает на входы $S0$, $S1$ СУАМ управляющие сигналы, которые через 20 нс ($S0$, $S1 \rightarrow Y$) выдадут на адресные входы *ЗУМК* новый адрес из регистра адреса (РА) СУАМ, в соответствии с которым новая команда поступает на входы *РМК*.

Условный переход на адрес вектора, инструкция CJV (путь 3). Первоначально расчетный путь информации совпадает с путем 2. УСА под управлением сигналов I и TST вырабатывает сигналы ME и PE, которые через логическую схему поступают на вход OE разрешения выдачи адреса вектора преобразователя адреса. С задержкой $OE \rightarrow Y$ новый адрес поступает на входы D СУАМ (сигналы на входах $S0$, $S1$ уже установлены) и через 20 нс ($D \rightarrow Y$) адрес поступает на адресные входы ЗУМК и новая микрокоманда поступает на РМК.

Переход по адресу из счетчика микрокоманд, инструкция CONT (путь 4). После прихода положительного фронта синхросигнала с задержкой $T \rightarrow Q$ сигналы микрокоманды поступают на входы I *УСА* и через 60 нс ($I \rightarrow S0$, $S1$) формируются сигналы $S0$, $S1$, вызывающие с задержкой 20 нс ($S0$, $S1 \rightarrow Y$) на выходы СУАМ адрес, находящийся в счетчике микрокоманд (СМК). В соответствии с этим адресом через время $A \rightarrow I$ из *ЗУМК* считывается новая микрокоманда, которая запишется в *РМК* следующим положительным фронтом синхросигнала, поданным через время установки (Установка D).

Как уже отмечалось, типовая схема микропроцессорного устройства обработки информации (см. рис. 7.41) состоит из двух частей: устройство микропрограммного управления и устройство обработки информации. Текущая микрокоманда, находящаяся в регистре микрокоманд, с одной стороны, управляет процессом обработки данных (цепочка *1—5*), с другой — формирует новую микрокоманду (цепочка *1—2—3—4*). Таким образом, в устройстве протекают два самостоятельных процесса, каждый из которых имеет свой цикл (циклы обработки данных и управления). Длительности этих циклов определяются быстродействием узлов устройства, функционирующих в цикле, числом узлов, а также характером исполняемых в каждой микрокоманде задач (т. е. видом путей прохождения информации). Выше были приведены примеры расчетов циклов обработки данных и циклов управления.

Для определения цикла работы всего микропроцессорного устройства необходимо в зависимости от решаемых устройством задач проанализировать все возможные пути прохождения информации в циклах управления и обработки данных, выявить наиболее критичные в каждом цикле и сравнить их. Максимальный из этих циклов и будет определять минимальный цикл работы микропроцессорного устройства обработки информации.

Пусть в 16-разрядном типовом микропроцессорном устройстве обработки информации для выполнения заданной программы наиболее длительным циклом обработки данных является цикл выполнения арифметических операций без сдвига с выдачей признака нуля результата на регистр состояния, а длительность цикла управления определяется выполнением инструкций УСА типа CONT, PUSH, JRP с условным переходом по адресу из регистра адреса СУАМ по значению величины регистра состояния. Тогда для микропроцессорного устройства имеем следующие циклы обработки данных и управления (табл. 7.48, рис. 7.49). Времена в табл. 7.48

Таблица 7.48. **Итоговая таблица задержек устройства обработки данных**

Цикл управления			Цикл обработки данных		
Узел	Путь в узле	Время, нс	Узел	Путь в узле	Время, нс
РМК	$T \to Q$	21	РМК	$T \to Q$	21
УСА	$I \to S0,\ S1$	60	МПС	$A,\ B \to G,\ P$	70
СУАМ	$S0,\ S1 \to Y$	20	СУП	$G,\ P \to C,\ Z$	15
ЗУМК	$A \to I$	70	СТМПС	$C0 \to Z$	52
РМК	Установка *D*	5	РС	Установка *D*	5
Итого	—	176	—	—	163

даны для 25 °С; длительность цикла управления превышает длительность цикла обработки данных, и, следовательно, минимальный рабочий цикл устройства будет равен 176 нс.

Рассмотрим использование микропроцессорного комплекта в различных вариантах организации процессорных структур. Выше было подробно описано применение микросхем серии К1804 для типовой схемы устройства обработки информации. Приведенная организация устройства рекомендуется для применения данного микропроцессорного комплекта и обеспечивает хорошее быстродействие. Однако данный микропроцессорный комплект можно использовать и в других вариантах построения систем. На рис. 7.50—7.54 представлены варианты структур, в которых может быть применен микропроцессорный комплект серии К1804, в том числе и рекомендуемая структура (рис. 7.50, структура 1). На всех рисунках устройство управления последовательностью микрокоманд *УУПМ* представляет собой сочетание УСА и СУАМ. Оно управляется сигналами микрокоманды по входу *I* и сигналами условия по входу *CC*, обеспечивая выполнение всех инструкций УСА с набором адресов, реализуемых СУАМ.

Тракты передачи, которые определяют быстродействие, показаны толстыми линиями со стрелками и с крестообразной штриховкой. Обозначения у трактов передачи информации имеют следующий смысл: *A* — адрес текущей микрокоманды (исполняемой); *A* + 1 — адрес следующей микрокоманды; (*A* — 1) — адрес предыдущей микрокоманды; *I* (*A*) — текущая микрокоманда,

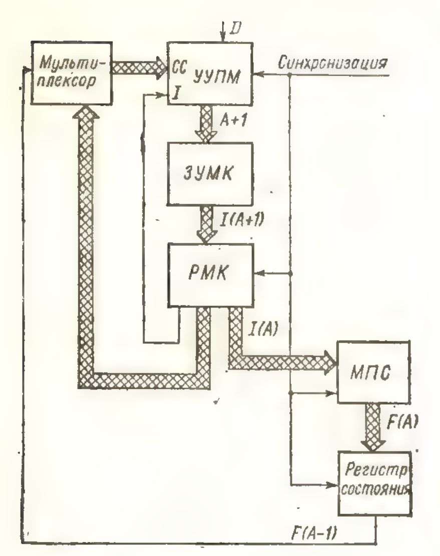


Рис. 7.50. Структура 1

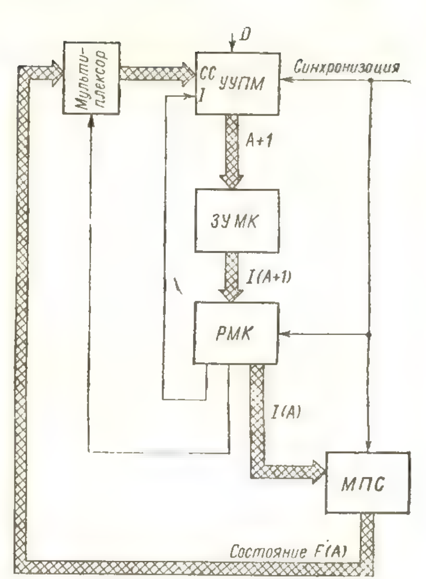


Рис. 7.51. Структура 2

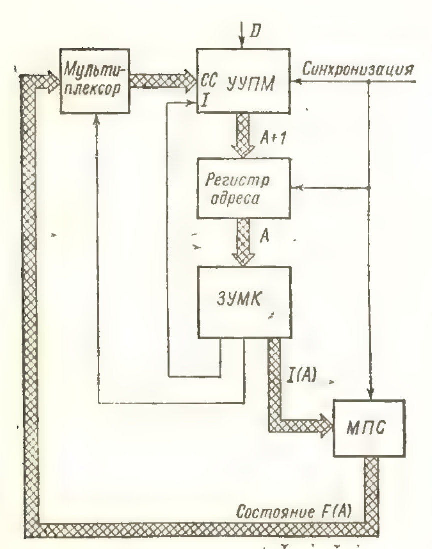


Рис. 7.52. Структура 3

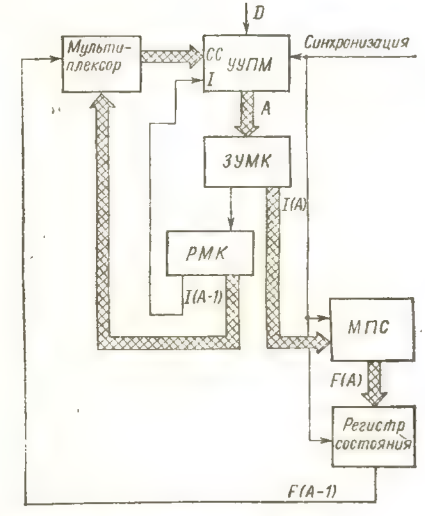


Рис. 7.53. Структура 4

выбранная по адресу A; F (A) — признаки состояний, полученные в результате исполнения микрокоманды, выбранной по адресу А.

В структуре 2 (рис. 7.51) регистр микрокоманд *РМК* содержит выполняемую микрокоманду. *ЗУМК* и микропроцессорные секции включены последовательно с точки зрения задержки распространения сигналов. Условные переходы выполняются в том же самом цикле, что и операция МПС, выдающая признаки состояния. В структуре 3 (рис. 7.52) регистр адреса на выходе *УУПМ* содержит адрес выполняемой микрокоманды. *ЗУМК* и МПС включены последовательно и определяют критичную длительность пути прохождения информации. Эта структура обеспечивает примерно то же быстродействие, что и структура 2, но требует меньше разрядов регистра, поскольку хранится только адрес (обычно 10 ... 12 бит), вместо хранения микрокоманды (обычно 40 ... 60 бит). В структуре 4 (рис. 7.53) регистр состояния обеспечивает информацию об условных переходах на основе результатов предшествующего цикла F (A — 1) и I (A — 1). *ЗУМК* и *МПС* здесь также включены последовательно. В структуре 5 (рис. 7.54) *РМК* содержит выполняемую микрокоманду, *регистр состояния* — результат предыдущей микрокоманды, *регистр адреса* — адрес следующей микрокоманды (A + 1). *УУПМ* вырабатывает адрес еще более новой микрокоманды (A + 2). Такая структура обеспечивает наиболее высокое быстродействие, но более сложна при программировании, поскольку выбор нужной микрокоманды должен происходить за две микрокоманды до ее исполнения.

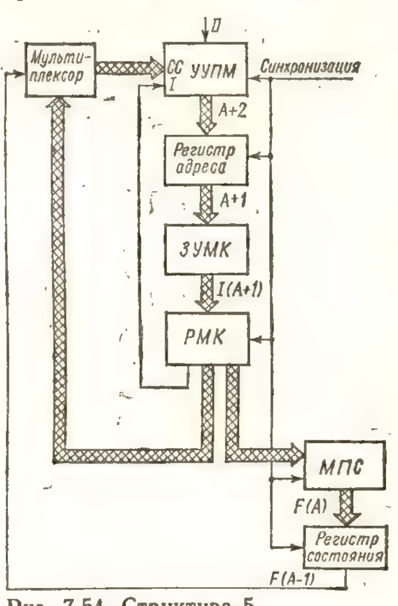


Рис. 7.54. Структура 5

При использовании микропроцессорного комплекта в различных вариантах организации структур длительности циклов будут определяться трактами передачи, показанными крестообразной штриховкой. При расчетах относительной длительности положительной и отрицательной частей цикла синхросигнала, наряду с ограничениями, накладываемыми минимальными длительностями сигналов низкого и высокого уровня для различных типов микросхем, необходимо учитывать времена установки ряда сигналов относительно фронтов синхроимпульса для МПС и СУАМ.

В качестве примера использования МПК БИС серии К1804 рассмотрим микроЭВМ с системой команд микроЭВМ «Электро-

ника-60». Основным устройством микроЭВМ является центральный процессор, который управляет магистралью при совместном ее использовании внешними устройствами и выполняет необходимые арифметико-логические операции по обработке информации. Структурная схема 16-разрядного центрального процессора показана на рис. 7.55.

В составе центрального процессора рассмотрим следующие функциональные блоки: *блок обработки данных* (БОД); *блок микропрограммного управления* (БМУ). Вся информация в центральном процессоре проходит через БОД, который управляется микропрограммно. Основные составные части БОД: арифметико-логическое устройство *АЛУ* со схемой ускоренного переноса *СУП*, мультиплексоры адреса регистров АЛУ, *мультиплексор входных данных*, *логика сдвигов*, *схема формирования признаков* и слова состояния процессора, *регистр команд*.

Главное устройство блока обработки данных — *АЛУ* построено на четырех четырехразрядных секциях К1804ВС1, объединенных схемой ускоренного переноса К1804ВР1.

Из шестнадцати 16-разрядных регистров *АЛУ* восемь являются программно-доступными и представляют собой регистры общего назначения процессора, а остальные регистры программно недоступны и используются для временного хранения информации в процессе выполнения микропрограмм. К этим шестнадцати регистрам можно обращаться по двум адресным входам (входы *A* и *B*), т. е. возможен независимый и одновременный доступ сразу к двум регистрам.

Восемь регистров общего назначения служат в качестве накопительных индексных регистров, регистров автоинкрементной и автодекрементной адресации и для других целей. Регистр *R6* используется как указатель стека, регистр *R7* служит счетчиком команд.

Все остальные устройства, входящие в состав БОД, выполнены на обычных логических интегральных схемах. Мультиплексоры и логика сдвигов реализуют как связи *АЛУ* с магистралью микроЭВМ, так и связи, необходимые в пределах БОД.

Блок БМУ включает в себя: *счетчик циклов*, запоминающее устройство микрокоманды, *регистр микрокоманд*, схему формирования адреса микрокоманды. БИС серии К1804 в микропрограммном устройстве управления применены в схеме формирования адреса микрокоманды. Основным элементом логики формирования адреса микрокоманды являются три четырехразрядные секции управления (две БИС К1804ВУ2 и одна БИС К1804ВУ1). Они формируют 12-разрядный адрес микрокоманды, используя информацию из следующих источников: *ПЛМ* или регистра микрокоманды; внутреннего регистра, собственного стека на 4 слова, запоминающего адреса возврата при выполнении подпрограмм. Кроме того, при формировании адреса микрокоманды возможен

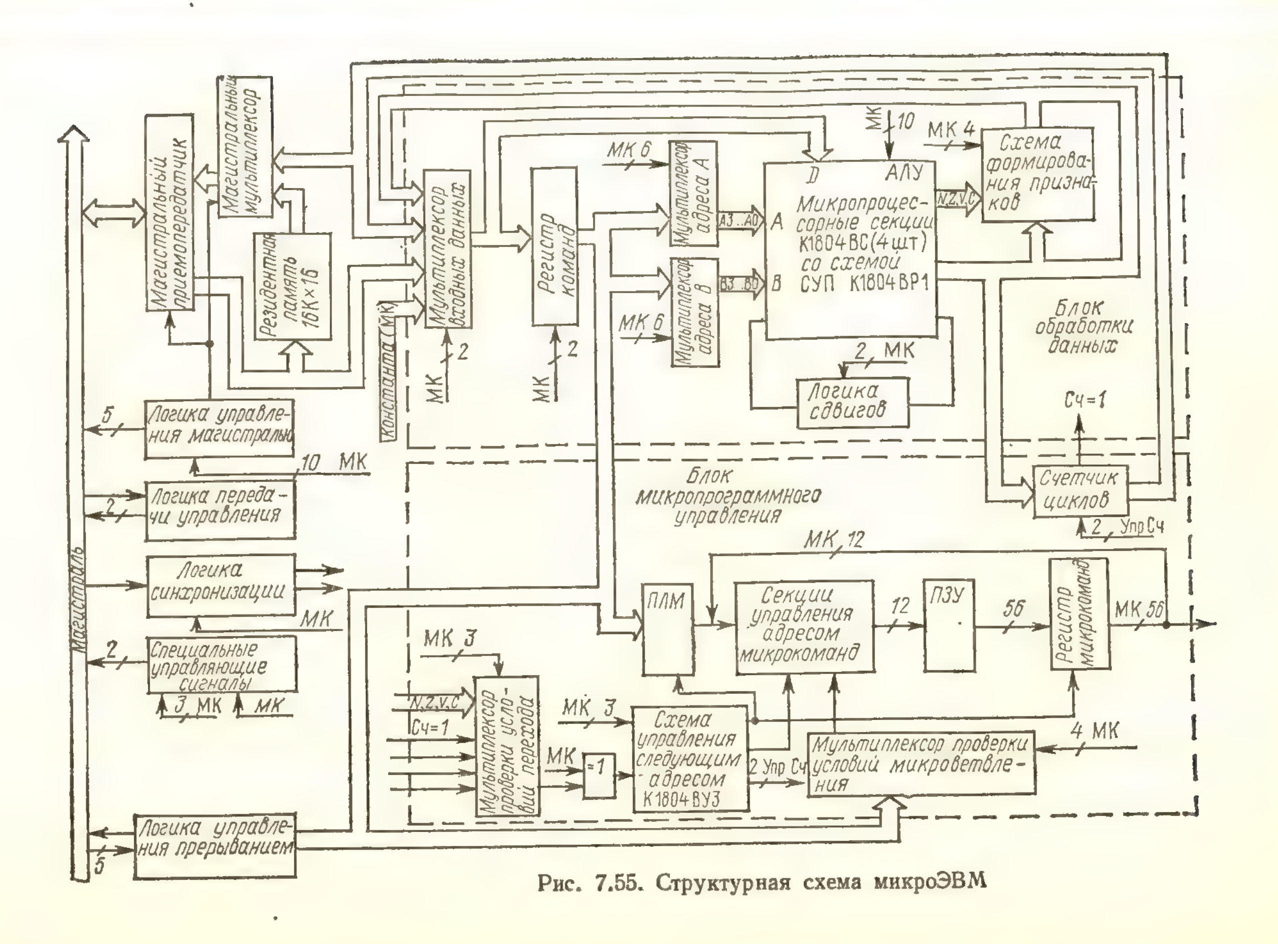


Рис. 7.55. Структурная схема микроЭВМ

естественный переход к следующей по номеру микрокоманде и зацикливание на текущей микрокоманде.

Работой секции формирования адреса микрокоманды управляет *схема управления следующим адресом* (БИС К1804ВУЗ). Адрес микрокоманды может быть сформирован либо от выходов *ПЛМ*, либо от выходов регистра микрокоманд. С помощью двух *ПЛМ*, к информационным входам которых подключаются выходы регистра команд, осуществляется дешифрация команд. Код на выходах *ПЛМ* определяет стартовый адрес микропрограмм, с помощью которых реализуется выполнение команд микроЭВМ.

Управление последовательностью выполнения микрокоманд (всевозможные ветвления, организация циклов) осуществляется с помощью мультиплексора проверки условий перехода, мультиплексора проверки условий микроветвления и счетчика циклов, реализованных на интегральных схемах средней степени интеграции.

Остальные узлы микроЭВМ (логика управления магистралью, логика синхронизации и т. д.) также реализованы на обычных интегральных схемах и поэтому подробно не рассматриваются.

В заключение приведем микросхемы серии К1804, не вошедшие в эту главу.

Микропроцессорные секции с расширенными возможностями К1804ВС2. Основными особенностями К1804ВС2 по сравнению с К1804ВС1 являются: выполнение специальных функций в АЛУ; выполнение логических и арифметических сдвигов сдвигателем АЛУ; наличие встроенных схем для реализации умножения, деления, нормализации, дополнения числа со знаком инкремента на единицу или на два, генерации паритета и размножения знака; двунаправленные шины; наличие сигналов фиксации положения секции в системе; возможность внешнего расширения РЗУ подсоединением любого числа дополнительных рабочих регистров и работе в двух- и трехадресном режимах.

Схема управления состоянием и сдвигами К1804ВР2. Предназначена для выполнения различных функций обслуживания микропроцессорных секций комплекта.

Схема управления последовательностью микрокоманд К1804ВУ4. Функционально эта схема заменяет три схемы К1804ВУ2 и одну К1804ВУЗ.

Четырехразрядные канальные приемопередатчики К1804ВУ1,-2,-3. Предназначены для подключения внутренних устройств микроЭВМ к общей шине (каналу).

Восьмиразрядный параллельный регистр К1804ИР2. Восьмиразрядный параллельный двунаправленный регистр К1804ИР3.

Системный тактовый генератор К1804ГГ1. Предназначен для синхронизации цифровых вычислительных устройств, построенных на базе МПК БИС серии К1804, и может быть использован для тактирования разнообразных цифровых устройств общего и специального назначения.

Глава 8

МИКРОПРОЦЕССОРНЫЙ КОМПЛЕКТ БИС СЕРИИ КР1802

8.1. ОБЩИЕ СВЕДЕНИЯ

МПК БИС серии КР1802 предназначен для построения одно- и многопроцессорных систем обработки информации средней и большой производительности. Включает БИС различного функционального назначения (процессорные, интерфейсные, ОЗУ), с помощью которых можно создавать вычислительные структуры различной разрядности. Гибкость системы микрокоманд и широкие функциональные возможности БИС комплекта позволяют эффективно реализовать практически любую систему команд или любые алгоритмы.

При разработке комплекта была использована «нарезка» по функциональному признаку. Гипотетический вычислительный модуль, содержащий основные узлы процессора, расчленялся на секции возможно большей разрядности с разнесением функциональных возможностей по БИС комплекта. Так, БИС АУ, имея 8-разрядную организацию, выполняет логические и арифметические операции над двумя операндами, а также сдвиг на один разряд вправо и влево. Однако БИС АУ лишена внутреннего регистрового ОЗУ (за исключением регистра расширения). Такое ОЗУ выполнено в виде отдельной БИС РОН. Расширяют вычислительные возможности комплекта две специализированные БИС: умножителя БИС УМН и арифметического расширителя (сдвигателя) БИС АР.

Комплект БИС этой серии включает интерфейсные схемы БИС И и БИС ИО, которые могут быть использованы для управления обменом информацией по магистрали с асинхронной дисциплиной обмена. Для микропрограммной реализации команд предусмотрены устройства хранения микропрограмм. Комплект дополнен БИС ПЛМ. БИС ПЛМ серии К556 представляет электрически программируемую логическую матрицу, реализующую 8 выходных функций от 16 входных переменных.

Нарезка по функциональному признаку затрудняет разработку аппаратуры. Так, возрастает число межсхемных пересылок, что усложняет организацию магистралей. В то же время повышаются функциональные возможности и обеспечивается гораздо бóльшая гибкость в разработке аппаратуры.

Комплект совместим со всеми микросхемами ТТЛ-типа и может быть использован совместно с МПК серий К589 и К1804.

МПК включает следующие БИС: арифметическое устройство (АУ); арифметический расширитель (АР); схему регистров общего

Таблица 8.1. **Состав МПК БИС серии КР1802**

Тип микросхемы	Обозначение	Средняя мощность потребления, мВт	Разрядность	Тип корпуса
КР1802ВС1	АУ	1000	8	2206.42-1
КР1802ВР1	АР	1000	16	
КР1802ВР2	УМН	1100	8	
КР1802ВВ1	ОИ	1000	Емкость 16 бит (4×4)	
КР1802ВВ2	И	900	—	
КР1802ИР1	РОН	600	Емкость 64 бит (16×4)	239.24-2

назначения (РОН); умножитель (УМН); схему обмена информацией (ОИ); схему интерфейса (И).

Обозначение микросхем серии КР1802 и отдельные характеристики приведены в табл. 8.1.

Свободные, не используемые в аппаратуре входы микросхем необходимо подключить к напряжению «1» или шине «Земля» в зависимости от логики работы микросхемы.

Инструмент для пайки и монтажа не должен иметь потенциал, превышающий 0,3 В относительно шины «Земля».

8.2. АРИФМЕТИЧЕСКОЕ УСТРОЙСТВО КР1802ВС1

БИС АУ представляет 8-разрядную схему с двухшинной структурой и предназначена для выполнения основного объема вычислений в процессорном блоке. БИС АУ принимает и обрабатывает два 8-разрядных операнда (при этом результат операции выдается одновременно на обе шины). БИС АУ имеет один внутренний регистр хранения промежуточных результатов.

БИС АУ позволяет выполнять: арифметическое сложение и вычитание двух операндов в дополнительном коде; логические операции И, ИЛИ, НЕ, сложение по модулю 2 (Исключающее ИЛИ); арифметические и логические сдвиги вправо и влево на один разряд, а также сдвиги двойных слов; операции маскирования. АУ формирует признаки результата операции.

БИС АУ обеспечивает двунаправленный обмен данными. Третье состояние на шинах *А* и *В* позволяет подключать схему непосредственно на общую двунаправленную магистраль. Однако двунаправленность шин *А* и *В* и связанное с нею временное мультиплексирование информации на этих шинах накладывают определенные ограничения на дисциплину обмена.

В состав БИС АУ входят (рис. 8.1) следующие устройства: буферы шин В и А с входными регистрами *БВ*, *БА*; усилители входных сигналов *УС*; дешифратор кода микрокоманды *ДШ*;

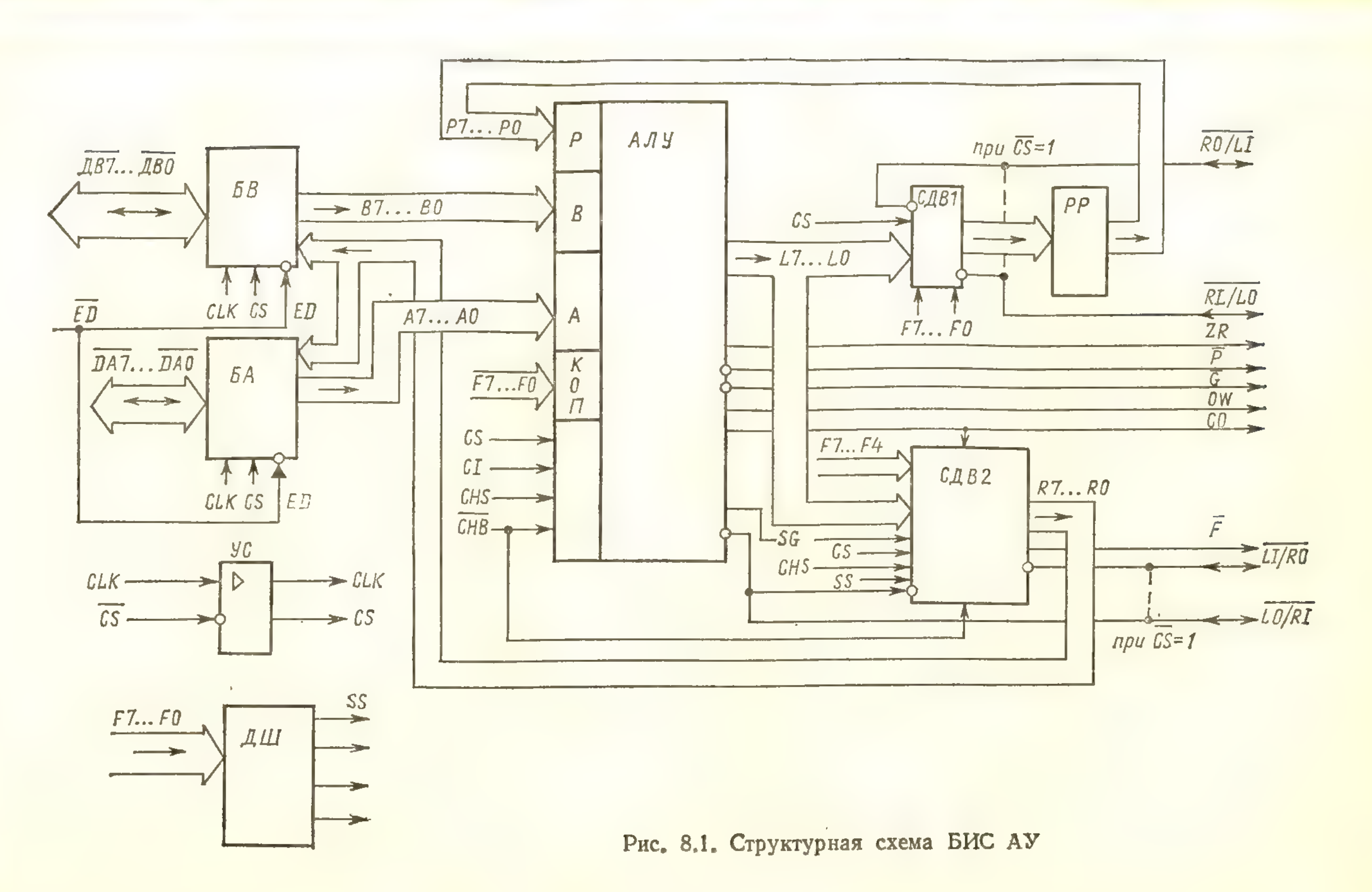


Рис. 8.1. Структурная схема БИС АУ

Таблица 8.2. Назначение выводов микросхемы КР1802ВС1

Номер вывода	Назначение	Разряд	Обозначение	Тип
1	Шина *A*	1	*DA*1	Двунаправленный
2	» *B*	2	*DB*2	
3	» *A*	2	*DA*2	
4	» *B*	3	*DB*3	
5	» *A*	3	*DA*3	
6	Шина микрокоманды	0	*F*0	Вход
7		1	*F*1	
8		2	*F*2	
9		3	*F*3	
10	Перенос	—	*CI*	
11	Общий	—	*GND*	—
12	Признак нуля	—	*ZR*	Выход
13	Синхронизация	—	*CLK*	
14	Шина микрокоманды	4	*F*4	Вход
15		5	*F*5	
16		6	*F*6	
17		7	*F*7	
18	Шина *A*	4	*DA*4	Двунаправленный
19	» *B*	4	*DB*4	
20	» *A*	5	*DA*5	
21	» *B*	5	*DB*5	
22	» *A*	6	*DA*6	
23	» *B*	6	*DB*6	
24	» *A*	7	*DA*7	
25	» *B*	7	*DB*7	
26	Переполнение	—	*OW*	Выход
27	Выход выдвигаемых разрядов	—	*F*	
28	Вывод сдвига СДВ2	—	*LO*/*RI*	Двунаправленный
29	Разрешение выдачи	—	*ED*	Вход
30	Выбор старшей БИС	—	*CHS*	
31	Инверсия знака	—	*CHB*	
32	Напряжение питания	—	U_{cc}	—
33	Перенос	—	*CO*	Выход
34	Сигнал распространения переноса	—	*P*	
35	Сигнал генерации переноса	—	*G*	
36	Выбор кристалла	—	*CS*	Вход
37	Вывод сдвига СДВ1	—	*RI*/*LO*	Двунаправленный
38		—	*RO*/*LI*	
39	Вывод сдвига СДВ2	—	*LI*/*RO*	
40	Шина *B*	0	*DB*0	
41	» *A*	0	*DA*0	
42	» *B*	1	*DB*1	

Примечание. Выходы *DA*7 ... *DA*0, *DB*7 ... *DB*0, *LO*/*RI*, *RI*/*LO*, *RO*/*LI*, *LI*/*RO* трехстабильные, выходы *ZR*, *OW*, *F* с открытым коллектором.

арифметико-логическое устройство *АЛУ*; схемы сдвига выходной информации *СДВ1* и *СДВ2*; регистр расширения *PP*.

В табл. 8.2 приведено назначение выводов БИС АУ. Между условным обозначением вывода и обозначением сигнала на этом выводе существует следующая связь: если вход или выход сигнала (информации) инверсный, то обозначение сигнала на выводе соответствует обозначению вывода с инверсией (например, $\overline{ED}$, $\overline{DB0}$). В противном случае обозначения вывода и сигнала совпадают. Все это относится и к описанию остальных БИС комплекта.

Условное графическое обозначение БИС АУ представлено на рис. 8.2.

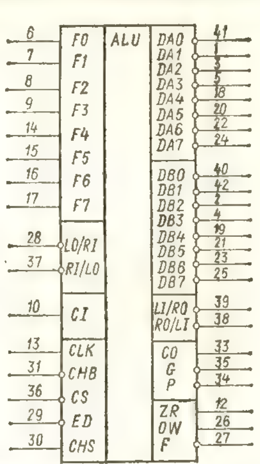


Рис. 8.2. Условное графическое обозначение БИС АУ

Операции, выполняемые АУ, определяются 8-разрядным кодом микрокоманды, поступающим на дешифратор.

Четыре разряда *F*3 ... *F*0 определяют собственно операции *АЛУ* над двумя операндами, а при маскировании модифицируют эти операции. Разряды *F*7 ... *F*4 определяют модификации операций *АЛУ*. Они управляют выбором одного из операндов *АЛУ*, т. е. в качестве операнда выбирают входную информацию буфера *БВ*, или содержимое регистра расширения *PP*, кроме этого, разрешают маскирование операндов *A* и *B* содержимым *PP*. Указанные разряды управляют схемой сдвига результата *АЛУ—СДВ2* и схемой сдвига регистра расширения *СДВ1* и записью в него.

В табл. 8.3 представлены операции АЛУ, где *A*, *B*, *P* — информация на соответствующих входах, а *L* — на выходе АЛУ; *A*7, *B*7, *P*7, *L*7 — старшие разряды информации. В графе $L = f(A, B)$ даны операции АЛУ над операндами *A* и *B*. Результат операции АЛУ есть функция *A* и *B* ($L = f(A, B)$). Разряды *F*4 ... *F*7 микрокоманды не должны принимать следующие значения: 0001 $\vee$ 0011 $\vee$ 1111. В графе $L = f(A, P)$ в качестве одного из операндов взят операнд *P*. Здесь разряды *F* 4 ... *F*7 должны иметь значения: 0001 $\vee$ 0011.

В следующей графе даны операции АЛУ, когда операнды *A* и *B* маскируются содержимым РР, т. е. результат операции АЛУ есть функция от *A*, *B* и *P* ($L = f(A, B, P)$). Здесь на входы *F*7 ... *F*4 следует подавать код 1111.

В табл. 8.3 даются также значения внутреннего признака вычитания *SS*, который формируется в ДШ и выдается в схему сдвига СДВ2. Только при операциях вычитания он равен единице, в остальных операциях он равен нулю.

Таблица 8.3. Операции БИС АУ

Мнемоника	Код микрокоманды				Признак SS	Операция	Результат операции АЛУ			
	$F0$	$F1$	$F2$	$F3$			$L = f(A, B)$	SG	$L = f(A, P)$	$L = f(A, B, P)$
КОН	0	0	0	0	0	Коньюнкция	$A \wedge B$	$\overline{A7 \wedge B7}$	$A \wedge P$	$A \wedge (B \vee \overline{P})$
ДКА	0	0	0	1	1	Декремент А	$A - 1 + CI$	$\overline{A7}$	$A - 1 + CI$	$A + \overline{B} \vee P + CI$
ИНА	0	0	1	0	1	Инверсия А	$\overline{A} + CI$	$\overline{A7}$	$\overline{A} + CI$	$\overline{A} \wedge P + B \wedge \overline{P} + CI$
ВЧК	0	0	1	1	1	Вычитание кодов	$A - B - 1 + CI$	$\overline{A7 \oplus B7}$	$A - P - 1 + CI$	$A - B \wedge P - 1 + CI$
Ф10	0	1	0	0	0	Функция 1 или 0	$CI - 1$	1	$CI - 1$	$A \vee P + B \wedge \overline{P} + CI$
ИНК	0	1	0	1	0	Инкремент	$A + CI$	$A7$	$A + CI$	$A + B \wedge \overline{P} + CI$
СЛ	0	1	1	0	0	Сложение	$A + B + CI$	$A7 \oplus B7$	$A + P + CI$	$A \vee \overline{P} + B \wedge P + CI$
СЛК	0	1	1	1	0	Сложение кодов	$A + B + CI$	$A7 \oplus B7$	$A + P + CI$	$A + B \wedge P + CI$
ИНВ	1	0	0	0	1	Инверсия В	$\overline{B} + CI$	$\overline{B7}$	$\overline{P} + CI$	$A \wedge \overline{P} + \overline{B} \wedge P + CI$
КНВ	1	0	0	1	0	Коньюнкция НЕ В	$A \wedge \overline{B}$	$\overline{A7} \vee B7$	$A \wedge \overline{P}$	$A \wedge (\overline{B \wedge P})$
ПА	1	0	1	0	0	Пересылка А	$A + CI$	$A7$	$A + CI$	$A \wedge P + B \wedge \overline{P} + CI$
ДИЗ	1	0	1	1	0	Дизьюнкция	$A \vee B$	$A7 \vee B7$	$A \vee P$	$A \vee B \wedge P$
ВЧ	1	1	0	0	1	Вычитание	$A - B - 1 + CI$	$\overline{A7 \oplus B7}$	$A - P - 1 + CI$	$A \wedge P + (\overline{B \wedge P}) + CI$
МД2	1	1	0	1	0	Сумма по модулю 2	$A \oplus B$	$\overline{A7 \oplus B7}$	$A \oplus P$	$A \oplus (B \wedge P)$
ПВ	1	1	1	0	0	Пересылка В	$B + CI$	$B7$	$P + CI$	$A \wedge \overline{P} + B \wedge P + CI$
ВЧО	1	1	1	1	1	Вычитание обратное	$B - A - 1 + CI$	$\overline{A7 \oplus B7}$	$P - A - 1 + CI$	$B \wedge P - A - 1 + CI$

Примечание. В графе $L = f(A, B)$ $F4 \ldots F7 \neq 0001 \vee 0011 \vee 1111$; в графе $L = f(A, P)$ $F4 \ldots F7 = 0001 \vee 0011$; в графе $L = f(A, B, P)$ $F4 \ldots F7 = 1111$.

Почти во всех операциях присутствует значение входного переноса CI, который обычно подается на младшую БИС АУ из микрокоманды. Подавая на вход переноса «1» или «0» можно получить на выходе БИС АУ соответственно коды «все 0» и «все 1», применяя операцию АЛУ Ф10. В других операциях (в зависимости от значения CI) можно получить инкремент или декремент операнда A или B. В операции ИНА получаем либо обратный код A (если CI = «0»), либо дополнительный код (если CI = «1»).

В графе SG показан результат логической операции со старшими разрядами A и B, который генерируется в АЛУ только при операциях над операндами A и B ($L = f(A, B)$) и поступает на схему сдвига СДВ2, где используется в модификациях операций.

В табл. 8.4 приведены модификации операций и значения признаков, вырабатываемых АУ при выполнении операций, задаваемых кодом $F7 \dots F4$. В графе $R7 \dots R0$ ($R7$ — старший разряд информации) дана информация на выходе схемы СДВ2, представляющая результат операции сдвига над данными $L7 \dots L0$ АЛУ. В четырех столбцах отражены значения соответствующих выходов СДВ2. В столбце $R0$ показано значение младшего разряда выхода СДВ2. Этот разряд может принимать три значения: $LI/R0$ — при всех операциях сдвига влево; $L1$ — при всех операциях сдвига вправо; $L0$, когда операции сдвига отсутствуют. В столбце $R6 \dots R1$ показана информация на выходах $R6 \dots R1$ схемы сдвига.

При всех операциях сдвига влево на них будет выдаваться $L5 \dots L0$, при всех сдвигах вправо — $L7 \dots L2$, а при отсутствии сдвигов — $L6 \dots L1$. В двух столбах показаны значения выхода $R7$ при CHS = «0» (кристалл не старший), а при CHS = «1» (кристалл старший). Значения $R7$ при CHS = «0» определяются так же как и значения $R0$: при всех операциях сдвига влево $R7$ принимает значение $L6$, при всех сдвигах вправо — значение $L0/RI$, при отсутствии сдвигов — значение $L7$.

Значения $R7$ в старшем кристалле (CHS = «1») определяются так же как и в младших кристаллах во всех модификациях, кроме следующих: САЛ, САП, РАП, СРП и РРП. Значение $R7$ в этих модификациях определяется самим смыслом модификации. Например, модификация «Сдвиг арифметический влево» должна сохранять знаковый (старший) разряд без изменения, поэтому $R7$ при CHS = «1» в этой модификации всегда принимает значение $L7$.

В графе «Информация, загруженная в РР» дана информация, записанная в регистр расширения РР при условии, если кристалл выбран, т. е. CS = «1». В трех столбцах показано, как информация с выхода АЛУ $L7 \dots L0$ и с выводов RO/LI, RI/LO при сдвигах влево, вправо или без сдвига записывается в РР (по переднему фронту CLK).

В графах $\overline{LO/RI}$, $\overline{RO/LI}$, $\overline{RI/LO}$, $\overline{LI/RO}$ показана информация на этих выводах в зависимости от кода $F7 \dots F4$. В тех модифика-

Таблица 8.4. Модификации операций БИС АУ и значения признаков

Мнемоника	Код модификации				Модификация	Операция АЛУ
	$F4$	$F5$	$F6$	$F7$		
САЛ	0	0	0	0	Сдвиг арифметический влево	$L = f(A, B)$
ОРС	0	0	0	1	Операция с сохранением РР	$L = f(A, P)$
ОЗР	0	0	1	0	Операция со словом с загрузкой РР	$L = f(A, B)$
ОРР	0	0	1	1	Операция с РР	$L = f(A, P)$
СЦЛ	0	1	0	0	Сдвиг циклический влево	$L = f(A, B)$
САП	0	1	0	1	Сдвиг арифметический вправо	
РЦЛ	0	1	1	0	Сдвиг циклический влево с РР	
РАП	0	1	1	1	Сдвиг арифметический вправо с РР	
СРП	1	0	0	0	Сдвиг, расширенный вправо	
СРЛ	1	0	0	1	Сдвиг, расширенный влево	
РРП	1	0	1	0	Сдвиг, расширенный вправо с РР	
РРЛ	1	0	1	1	Сдвиг, расширенный влево с РР	
СЦП	1	1	0	0	Сдвиг циклический вправо	
ОПС	1	1	0	1	Операция со словом	
РЦП	1	1	1	0	Сдвиг циклический вправо с РР	
ОПМ	1	1	1	1	Операция маскирования	$L = f(A, B, P)$

Результат операции САЛУ R7 ... R0				Информация, загруженная в РР (CS = 1)		
R7, CHS = 1	R7, CHS = 0	R6 ... R1	R0	P7	P6 ... P1	P0
L7	*L6*	*L5 ... L0*	*LI/RO*	Содержимое РР не изменяется		
L7	*L7*	*L6 ... L1*	*L0*	То же		
L7	*L7*	*L6 ... L1*	*L0*	*L7*	*L6 ... L1*	*L0*
L7	*L7*	*L6 ... L1*	*L0*	*L7*	*L6 ... L1*	*L0*
L6	*L6*	*L5 ... L0*	*LI/RO*	Содержимое РР не изменяется		
CHB⊕SG	*LO/RI*	*L7 ... L2*	*L1*	То же		
L6	*L6*	*L5 ... L0*	*LI/RO*	*P6*	*P5 ... P0*	*RO/RI*
CHB⊕SG	*LO/RI*	*L7 ... L2*	*L1*	*RI/LO*	*P7 ... P2*	*P1*
CHB⊕CO	*LO/RI*	*L7 ... L2*	*L1*	Содержимое РР не изменяется		
L6	*L6*	*L5 ... L0*	*LI/RO*	То же		
CHB⊕CO	*LO/RI*	*L7 ... L2*	*L1*	*RI/LO*	*P7 ... P2*	*P1*
L6	*L6*	*L5 ... L0*	*LI/RO*	*P6*	*P5 ... P0*	*RO/LI*
LO/RI	*LO/RI*	*L7 ... L2*	*L1*	Содержимое РР не изменяется		
L7	*L7*	*L6 ... L1*	*L0*	То же		
LO/RI	*LO/RI*	*L7 ... L2*	*L1*	*RI/LO*	*P7 ... P2*	*P1*
L7	*L7*	*L6 ... L1*	*L0*	Содержимое РР не изменяется		

Мнемоника	$\overline{LO/RI}$			$\overline{RO/LI}$		$\overline{RI/LO}$	
	$CS \wedge CHS=1$	$CS \wedge \overline{CHS}=1$	$CS=0$	$CS=1$	$CS=0$	$CS=1$	$CS=0$
САЛ	1	$\overline{L7}$	$\overline{LI/RO}$	Вход		RO/LI	RO/LI
ОРС	Z	Z	Z	Z	Z	Z	Z
ОЗР	Z	Z	Z	Z	Z	Z	Z
ОРР	Z	Z	Z	Z	Z	Z	Z
СЦЛ	$\overline{L7}$	$\overline{L7}$	$\overline{LI/RO}$	Вход		RO/LI	RO/LI
САП	Вход			$\overline{RI/LO}$	$\overline{RI/LO}$	Вход	
РЦЛ	$\overline{L7}$	$\overline{L7}$	$\overline{LI/RO}$	Вход		$\overline{P7}$	$\overline{RO/LI}$
РАП	Вход			$\overline{P0}$	$\overline{RI/LO}$	Вход	
СРП	Вход			RI/LO	RI/LO	Вход	
СРЛ	$\overline{CHB \oplus CO}$	$\overline{L7}$	$\overline{LI/RO}$	Вход		$\overline{RO/LI}$	$\overline{RO/LI}$
РРП	Вход			$\overline{P0}$	$\overline{RI/LO}$	Вход	
РРЛ	$\overline{CHB \oplus CO}$	$\overline{L7}$	$\overline{LI/RO}$	Вход		$\overline{P7}$	$\overline{RO/LI}$
СЦП	Вход			$\overline{RI/LO}$	$\overline{RI/LO}$	Вход	
ОПС	Z	Z	Z	Z	Z	Z	Z
РЦП	Вход			$\overline{P0}$	$\overline{RI/LO}$	Вход	
ОПМ	Z	Z	Z	Z	Z	Z	Z

Продолжение табл. 8.4

$\overline{LI/RO}$		Значения признаков		
$CS=1$	$CS=0$	$\overline{F}$	OW	ZR
Вход		$\overline{L6}$	$(C0\oplus C6)\vee(L7\oplus L6)$	
$C3$	$C3$	$\overline{C0}\oplus SS$	$C0\oplus C6$	
$C3$	$C3$	То же	То же	
$C3$	$C3$	То же	То же	
Вход		$\overline{L7}$	$(C0\oplus C6)\vee(L7\oplus L6)$	
$\overline{L0}$	LO/RI	$\overline{LO/RI}$	0	
Вход		$\overline{L7}$	$(C0\oplus C6)\vee(L7\oplus L6)$	
$\overline{L0}$	$\overline{LO/RI}$	$\overline{LO/RI}$	0	$\bigvee_{i=0}^{7} Ri$
$\overline{L0}$	$\overline{O/RI}$	$\overline{O/RI}$	0	
Вход		$\overline{L7}$	$(C0\oplus C6)\vee(L7\oplus L6)$	
$\overline{L0}$	$\overline{LO/RI}$	$\overline{LO/RI}$	0	
Вход		$\overline{L7}$	$(C0\oplus C6)\vee(L7\oplus L6)$	
$\overline{L0}$	$\overline{LO/RI}$	$\overline{LO/RI}$	0	
$C3$	$C3$	$\overline{C0}\oplus SS$	$C0\oplus C6$	
$\overline{L0}$	$\overline{LO/RI}$	$\overline{LO/RI}$	0	
$C3$	$C3$	$\overline{C0}\oplus SS$	$C0\oplus C6$	$\bigvee_{i=0}^{7} Ri\wedge Pi$

циях, где нет сдвига, все выводы, кроме LI/RO, находятся в третьем состоянии (в таблице это обозначено Z). На вывод LI/RO выдается значение переноса $C3$ из третьего разряда АЛУ, которое может быть использовано дополнительной аппаратурой при реализации операций десятичной арифметики.

Графы $\overline{LO/RI}$, $\overline{RO/LI}$, $\overline{RI/LO}$, $\overline{LI/RO}$ разбиты на столбцы, один из которых указывает информацию на соответствующем выводе при CS = «0» (кристалл не выбран). Если CS = «0», то схемы сдвига «напрямик» соединяют соответствующие выводы, т. е. LO/RI с LI/RO и RI/LO с RO/LI. Информация на этих выводах определяется выводом, который является выходом в данной операции. Например, в операции РЦЛ выходом является вывод LO/RI, на который выводится значение $L7$. Этот вывод внешним монтажом подключен на вывод RO/LI, который «напрямик» соединен с выводом RI/LO и уже с этого вывода внешним монтажом на LI/RO.

В графе $\overline{LO/RI}$ (столбец $CS \wedge CHS$ = «1») дана информация на выводе LO/RI в старшем и выбранном кристалле.

Признаки $\overline{F}$ и OW вырабатываются только в выбранном и старшем кристалле, т. е. $CS \wedge CHS$ = «1». Если $CS \wedge \wedge \overline{CHS}$ = «1», то на выводах F и OW напряжение высокого уровня.

Признак ZR формируется только в выбранном кристалле, если кристалл не выбран (CS = «0»), то ZR = «1».

Сигнал переполнения при всех сдвигах влево вырабатывается как функция ИЛИ сигнала переполнения в операции АЛУ ($CO \oplus C6$) и сигнала переполнения при сдвиге влево ($L7 \oplus L6$), где CO — перенос из старшего 7-го разряда, $C6$ — перенос из 6-го разряда.

Сигнал ZR указывает на нулевой результат СДВ1. Если выполняется маскирование, то на нуль анализируется коньюнкция результата схемы сдвига и содержимого РР, т. е. $ZR = \mathrm{R0} \wedge \wedge \mathrm{P0} \vee \mathrm{R1} \wedge \mathrm{P1} \ldots \vee \mathrm{R7} \wedge \mathrm{P7}$.

На вывод F в модификациях со сдвигом влево выдается значение выдвигаемого разряда в обратном коде, при сдвигах вправо — значение $\overline{LO/RI}$. При отсутствии сдвигов на этот вывод выдается сигнал $\overline{CO} \oplus SS$.

Поясним содержание таблицы на примере модификации «Сдвиг арифметический влево». В результате выполнения этой операции знаковый (старший) разряд должен оставаться на месте, сдвигаются влево только значащие разряды, в освобождающиеся разряды задвигаются нули.

Рассмотрим графу $R7 \ldots R0$. В ней в столбце $R7$, CHS = «1» дано значение $L7$, указывающее, что в старшем кристалле знаковый разряд не сдвигается. В столбце $R7$, CHS = «0» $L6$ означает, что в остальных БИС АУ разряды результата АЛУ сдвигаются влево на один разряд.

В столбце $R0$ указано значение вывода LI/RO.

Рассмотрим графу $\overline{LO/RI}$. Если кристалл выбран и он старший, то на выводе LO/RI присутствует «1», поступающая на вывод RO/LI, поскольку вывод в операции «напрямик» соединен с выводом RI/LO, то «1» присутствует на выводе RI/LO, а значит и на выводе LI/RO. Так как этот вывод с инверсией, то в разряд $R0$ выдвигается инверсия «1», т. е. «0», что и отображается в столбце $R0$. В этой операции вывод LI/RO и RO/LI работают как входы. Операция не изменяет РР. Признак переполнения OW здесь формируется следующим образом: $(CO \oplus C6) \vee (L7 \oplus L6)$.

На рис. 8.3 представлена временна́я диаграмма работы БИС АУ. Для обеспечения правильного функционирования БИС на вход CLK необходимо подать импульсы синхронизации с периодом t_{CY}. На положительном импульсе синхронизации необходимо подать: код микрокоманды $F7 \ldots F0$ и данные по шинам DA и DB. При положительном уровне импульса синхронизации входные данные, поступившие по шинам DB и DA, появляются соответственно на выходах регистров БВ и БА, а по отрицательному фронту — защелкиваются. На вход Р АЛУ данные поступают с РР. Код микрокоманды можно изменять только на следующем положительном синхроимпульсе. Входные данные на шинах DA и DB можно снимать через время удержания t_H после отрицательного фронта импульса синхронизации. Для выполнения АЛУ соответствующих действий на входы CS, CI, CHB и CHS подаются соответствующие сигналы на отрицательном импульсе синхронизации за время установки t_{SU} этих сигналов до положительного фронта импульса синхронизации.

При наращивании БИС АУ необходимо выдерживать время установки сигналов на входах переносов RO/LI, RI/LO до положительного фронта сигнала синхронизации.

После освобождения шин DB и DA от данных подается (если необходимо считать данные с АУ) сигнал на вход ED, переводящий буферы шин БВ и БА в выходной режим работы.

На внешние шины выдается результат операции АУ. Признаки результата операции формируются через время распространения t_P от отрицательного фронта синхроимпульса. Данные на шинах сбрасываются либо по положительному фронту сигнала $\overline{ED}$, либо по положительному фронту синхроимпульса через соответствующее время распространения. По положительному фронту синхроимпульса сбрасываются и признаки операции. После соответствующего времени удержания от положительного фронта синхроимпульса можно сбрасывать код микрокоманды с соответствующими сигналами на входах CI и CS. После этого начинают следующий цикл. Результат операции, выполненной в АЛУ, может быть записан в РР для использования в последующих микрокомандах.

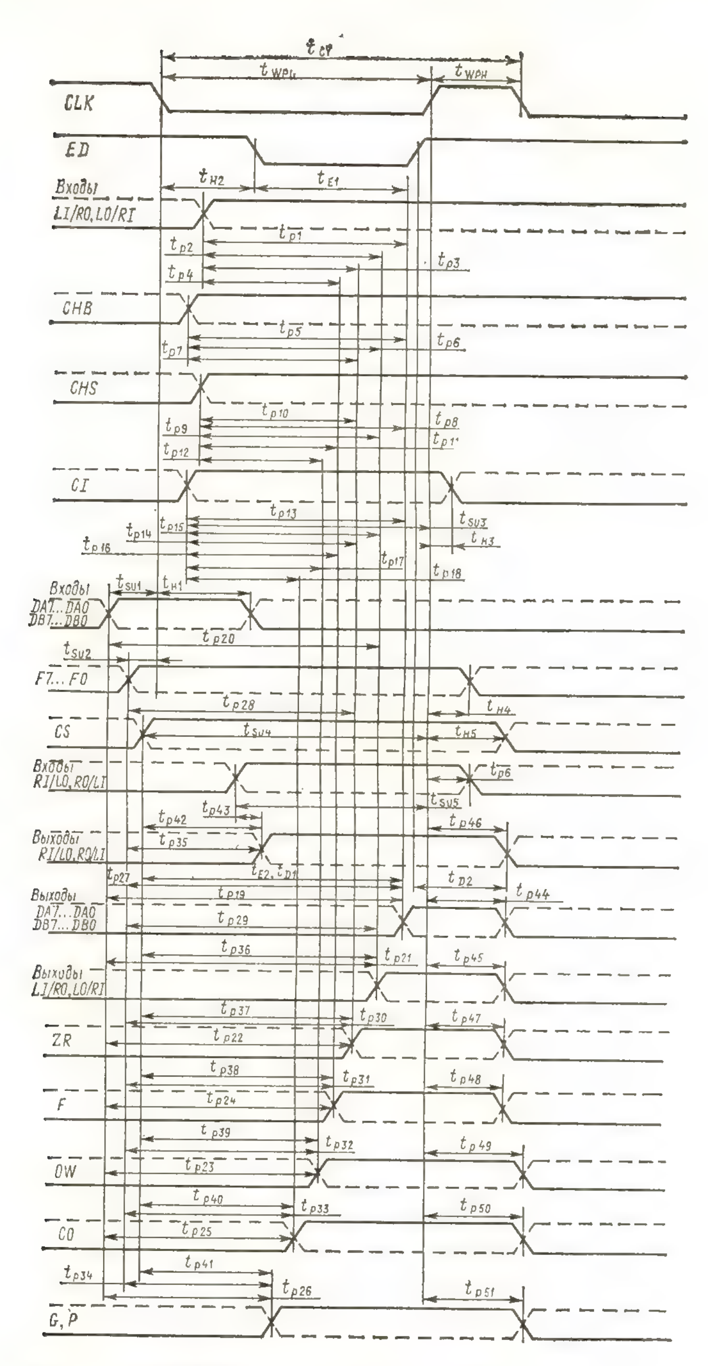


Рис. 8.3. Временна́я диаграмма работы БИС АУ

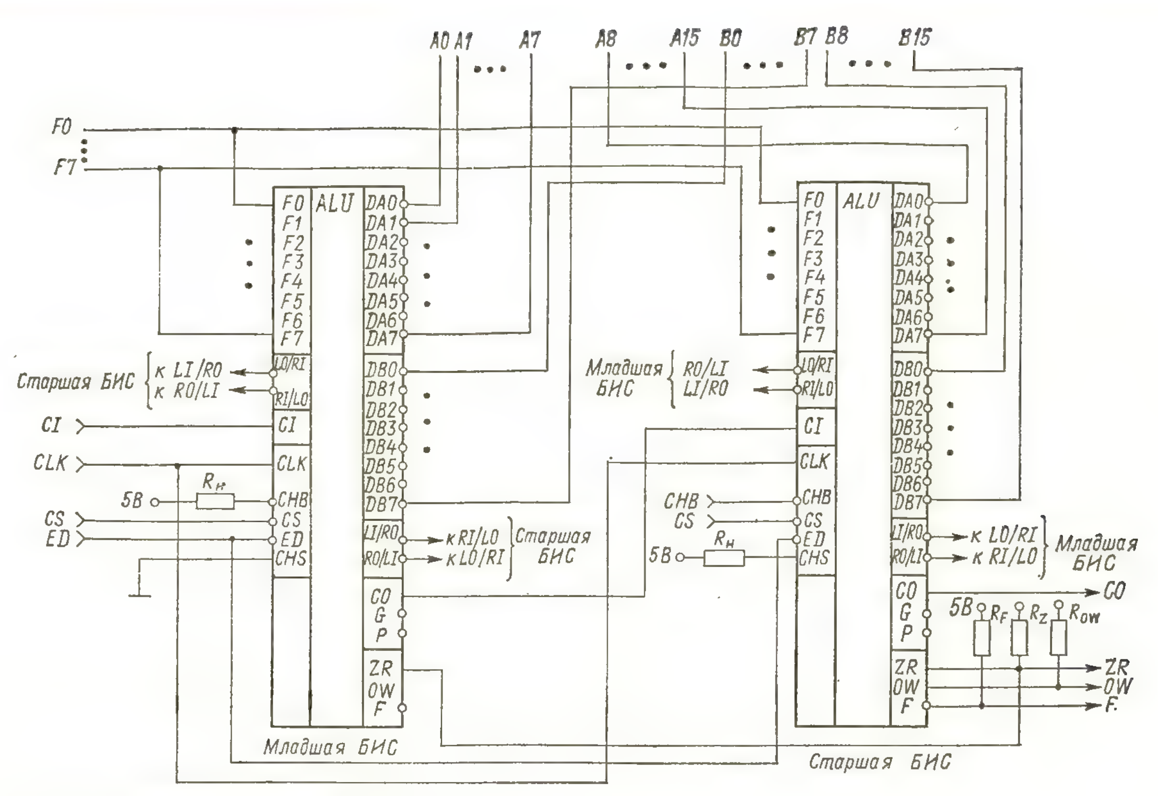


Рис. 8.4. Схема наращивания БИС АУ

Таблица 8.5. **Статические параметры микросхемы КР1802ВС1**

Параметр, единица измерения	Обозначение	Значения параметров		Режим измерения
		мин.	макс.	
Входной ток «0», мА, для входов:	I_{IL}			$U_{CC} = 5{,}25$ В $U_{IL} = 0 \dots 0{,}5$ В $U_{IH} = 2{,}4 \dots 4{,}5$ В
F0, ED, CLK, DA7 ... DA0, DB7 ... DB0		–0,25	—	
F7 ... F5, F3 ... F1		–0,5	—	
F4		–0,75	—	
CHB, LI/RO, RI/LO, RO/LI		–1,0	—	
CS, CHS		–0,4	—	
LO/RI		–2,0	—	
CI		–1,6	—	
Входной ток «1», мкА, для входов:	I_{IH}			$U_{CC} = 5{,}25$ В $U_{IH} = 5{,}25$ В
F0, ED, CLK, CS, CHS, CI DA7 ... DA0, DB7 ... DB0, F7 ... F5, F3 ... F1, CHB,		—	40	
LI/RO, RI/LO, RO/LI		—	80	
F4		—	120	
LO/RI		—	160	
Выходной ток «1», мкА, для выходов *F, OW, ZR*	I_{OH}	—	100	$U_{CC} = 4{,}75$ В $U_{TH} = 2$ В $U_{TL} = 0{,}8$ В
Выходное напряжение «0», В, по выходам:	U_{OL}			$U_{CC} = 4{,}75$ В $U_{TH} = 2$ В
DA7 ... DA0, DB7 ... DB0, G		—	0,5	$I_{OL} = 15$ мА
CO, F, ZR, OW, P		—	0,5	$I_{OL} = 10$ мА
LI/RO, LO/RI, RO/LI, RI/LO		—	0,5	$I_{OL} = 6$ мА $U_{TL} = 0{,}8$ В
Выходное напряжение «1», В, по выходам:	U_{OH}			$U_{CC} = 4{,}75$ В $U_{TH} = 2$ В
DA7 ... DA0, DB7 ... DB0, C0, P, G		2,4	—	$I_{OH} = -1$ мА
LI/RO, LO/RI, RO/LI, RI/LO		2,4	—	$I_{OH} = -0{,}6$ мА $U_{TL} = 0{,}8$ В
Ток выключенного состояния, мкА, по выходам:	I_{OZ}			$U_{CC} = 5{,}25$ В $U_{OH} = 5{,}25$ В $U_{TH} = 2$ В $U_{TL} = 0{,}8$ В
DA7 ... DA0, DB7 ... DB0		—	150	
LO/RI, RO/LI, RI/LO, LI/RO		—	250	
Ток короткого замыкания, мА, по всем выходам, кроме *F, OW, ZR*	I_{OS}	–90	–10	$U_{CC} = 5{,}0$ В $U_{OL} = 0$ В
Ток потребления, мА	I_{CC}	—	250	$U_{CC} = 5{,}25$ В $U_{IL} = 0 \dots 0{,}5$ В

Таблица 8.6. **Динамические параметры микросхемы КР1802ВС1**

Параметр	Обозначение	Значения, нс		
		мин.	ном.	макс.
Длительность цикла	t_{CY}	200	140	—
Длительность положительного синхроимпульса	t_{WPH}	60	40	—
Длительность отрицательного синхроимпульса	t_{WPL}	140	100	—
Время установки сигнала относительно отрицательного фронта синхроимпульса на входах:				
DA7 ... DA0, DB7 ... DB0	t_{SU1}	15	—	—
F7 ... F0	t_{SU2}	30	—	—
Время установки сигнала относительно положительного фронта синхроимпульса на входах:				
CI	t_{SU3}	45	—	—
CS	t_{SU4}	25	—	—
RO/LI, RI/LO	t_{SU5}	10	—	—
Время удержания сигнала относительно отрицательного фронта синхроимпульса на входах:				
DA7 ... DA0, DB7 ... DB0	t_{H1}	30	—	—
ED	t_{H2}	10	—	—
Время удержания сигнала относительно положительного фронта синхроимпульса на входах:				
CI	t_{H3}	10	—	—
F7 ... F0	t_{H4}	20	—	—
CS	t_{H5}	30	—	—
RO/LI, RI/LO	t_{H6}	30	—	—
Время задержки распространения сигнала от входов *LI/RO, LO/RI* до выходов:				
DA7 ... DA0, DB7 ... DB0	t_{p1}	—	20	37
LO/RI, LI/RO	t_{p2}	—	18	32
ZR	t_{p3}	—	30	45
F	t_{p4}	—	20	30
То же от входа *CHB* до выходов:				
DA7 ... DA0, DB7 ... DB0	t_{p1}	—	38	54
LO/RI	t_{p6}	—	36	50
ZR	t_{p7}	—	48	68
То же от входа *CHS* до выходов:				
DA7 ... DA0, DB7 ... DB0	t_{p8}	—	50	70
LO/RI	t_{p9}		48	67
ZR	t_{p10}		60	85
F	t_{p11}		50	70
OW	t_{p12}		35	50
То же от входа *CI* до выходов:				
DA7 ... DA0, DB7 ... DB0	t_{p13}	—	60	85
ZR	t_{p14}	—	68	95
LI/RO, LO/RI	t_{p15}	—	58	80
F	t_{p16}	—	60	85
OW	t_{p17}	—	30	42
CO	t_{p18}	—	16	23

Продолжение табл. 8.6

Параметр	Обозначение	Значения, нс		
		мин.	ном.	макс.
То же от входов *DA*7 ... *DA*0, *DB*7 ... *DB*0 до выходов:				
*DA*7 ... *DA*0, *DB*7 ... *DB*0	t_{p19}	—	110	155
LI/*RO*, *LO*/*RI*	t_{p20}	—	108	150
переноса из младшей тетрады на выход *LI*/*RO*	t_{p21}	—	100	140
ZR	t_{p22}	—	120	168
OW	t_{p23}	—	115	160
F	t_{p24}	—	110	155
CO	t_{p25}	—	100	140
G, *P*	t_{p26}	—	105	147
То же от входов *F*7 ... *F*0 до выходов:				
*DA*7 ... *DA*0, *DB*7 ... *DB*0	t_{p27}	—	130	182
LI/*RO*, *LO*/*RI*	t_{p28}	—	128	180
переноса из младшей тетрады на выход *LI*/*RO*	t_{p29}	—	120	168
ZR	t_{p30}	—	140	200
F	t_{p31}	—	130	182
OW	t_{p32}	—	135	190
CO	t_{p33}	—	120	168
G, *P*	t_{p34}	—	125	175
RO/*LI*, *RI*/*LO*	t_{p35}	—	130	182
То же от входа *CS* до выходов:				
LI/*RO*, *LO*/*RI*	t_{p36}	—	45	63
ZR	t_{p37}	—	70	98
F	t_{p38}	—	50	70
OW	t_{p39}	—	40	56
CO	t_{p40}	—	25	35
G, *P*	t_{p41}	—	30	42
RO/*LI*, *RI*/*LO*	t_{p42}	—	30	45
То же от входов *RO*/*LI*, *RI*/*LO* до выходов *RI*/*LO*, *RO*/*LI*	t_{p43}	—	18	25
То же от положительного фронта синхроимпульса *CLK* до выходов:				
*DA*7 ... *DA*0, *DB*7 ... *DB*0	t_{p44}	—	125	175
LI/*RO*, *LO*/*RI*	t_{p45}	—	123	172
RO/*LI*, *RI*/*LO*	t_{p46}	—	65	90
ZR	t_{p47}	—	135	190
F	t_{p48}	—	125	175
OW	t_{p49}	—	130	182
CO	t_{p50}	—	110	155
G, *P*	t_{p51}	—	115	160
Время включения до выходов *DA*7 ... *DA*0, *DB*7 ... *DB*0 от входов:				
ED	t_{E1}	—	30	42
CS	t_{E2}	—	45	63
Время выключения до выходов *DA*7 ... *DA*0, *DB*7 ... *DB*0 от входов:				
CS	t_{D1}	—	45	63
ED	t_{D2}	—	30	42

В табл. 8.5 и 8.6 приведены статические и динамические параметры БИС АУ.

На рис. 8.4 приведен пример объединения БИС АУ при наращивании.

8.3. АРИФМЕТИЧЕСКИЙ РАСШИРИТЕЛЬ КР1802ВР1

БИС КР1802ВР1 является арифметическим расширителем (АР), работающим с 16-разрядной информацией, и предназначена для расширения функциональных возможностей процессора. АР позволяет за один такт осуществлять арифметические, логические, циклические сдвиги влево и вправо, а также сдвиги со «склейкой результата. В АР предусмотрена операция определения кода (номера) первой слева единицы. При необходимости на АР можно сдвигать информацию и больше 16 разрядов (32, 48 и т. д.), но для этого требуется несколько тактов.

Условное графическое обозначение АР представлено на рис. 8.5, назначение выводов — в табл. 8.7.

Рис. 8.5. Условное графическое обозначение БИС АР

Структурная схема АР (рис. 8.6) включает следующие устройства:

буфер данных *БД* с входным регистром; обеспечивает прием 16-разрядной информации на свой внутренний регистр и выдачу результата операции над этим операндом;

сдвигатель *СД*; выполняет сдвиг 16-разрядного числа на любое число разрядов от 1 до 15. Число выдвигаемых разрядов определяется параметром сдвига *П3* ... *П*0. При сдвиге влево старший разряд параметра сдвига *S4* = 0, а значение *П3* ... *П0* подается в прямом коде. При сдвиге вправо *S4* = 1, а *П3* ... *П0* подается в дополнительном коде. С выхода *СД* выдвигаемая информация *K15* ... *K0* поступает на регистр расширения РР. СД также определяет номер разряда первой единицы слева;

регистр расширения *РР*; предназначен для хранения выдвигаемых разрядов. С помощью *РР* обеспечивается сдвиг слов, больших 16 разрядов;

блок склейки результата *БСР*; служит для формирования результата в зависимости от сдвига и представляет собой мультиплексор, выполняющий операцию логического сложения;

схему выработки нуля *СВН*; анализирует на равенство нулю все разряды *R*15 ... *R*0;

схему выработки переполнения *СВП*; генерирует признак переполнения при арифметических сдвигах влево, если хотя бы

один из выдвигаемых разрядов не совпал со старшим (знаковым) разрядом;

мультиплексор параметра сдвига *МПС*; пропускает на вход регистра параметра сдвига *РПС* параметр сдвига с двунаправленной магистрали *SHB* или с шины *SHI* в зависимости от управляющего сигнала выбора параметра *SHS*;

регистр параметра сдвига *РПС*; обеспечивает хранение параметра сдвига в момент выдачи на двунаправленную магистраль;

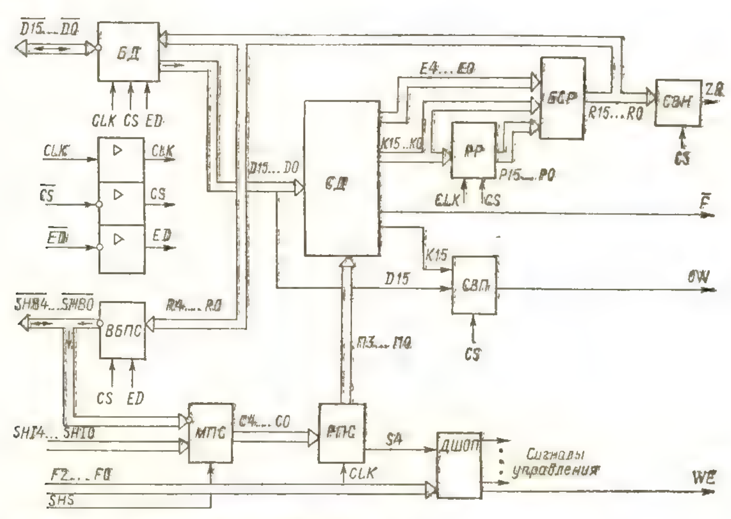


Рис. 8.6. Структурная схема БИС АР

выходной буфер параметра сдвига *ВБПС*; служит для выдачи результата операции поиска левой единицы на магистраль *SHB*;

дешифратор кода операции микрокоманды *ДШОП*.

В табл. 8.8 даны операции АР с признаками, вырабатываемыми в этих операциях. Каждая операция задается 3-разрядным кодом $F2 \dots F0$ и значением старшего разряда параметра сдвига $S4$. Операции СЛЛСР, СЛПСР, САЛСР при значении $S4$, противоположном смыслу операции, вырождаются в операции СЛП, СЛЛ и САП соответственно, но с признаком $WE = 0$.

В графе «Результат операции» дана информация на выходе *БСР*. Графа разбита на два столбца. В столбце $R(N-1) \dots R0$ $(N \neq 0)$ — информация, представляющая вдвигаемые разряды (при сдвиге влево), а в столбце $R15 \dots RN$ — информация после сдвига (если сдвиг производится со склейкой результата, то эта информация логически складывается с содержимым *РР*).

При сдвиге вправо в столбце $R15 \dots RN$ показаны вдвинутые разряды, а в столбце $R(N-1) \dots R0$ $(N \neq 0)$ — сдвинутая исходная информация.

Таблица 8.7. **Назначение выводов микросхемы КР1802ВР1**

Номер вывода	Назначение	Разряд	Обозначение	Тип
1	Общий	—	*GND*	—
2	Шина данных	0	*D*0	Двунаправленный
3		1	*D*1	
4		2	*D*2	
5		3	*D*3	
6		4	*D*4	
7		5	*D*5	
8		6	*D*6	
9		7	*D*7	
10		8	*D*8	
11	Общий	—	*GND*	—
12	Шина данных	9	*D*9	Двунаправленный
13		10	*D*10	
14		11	*D*11	
15		12	*D*12	
16		13	*D*13	
17		14	*D*14	
18		15	*D*15	
19	Признак равенства нулю	—	*ZR*	Выход
20	Синхронизация	—	*CLK*	Вход
21	Признак переполнения	—	*OW*	Выход
22	Общий	—	*GND*	—
23	Последний выдвинутый разряд или признак наличия единицы	—	*F*	Выход
24	Разрешение выдачи информации	—	*ED*	Вход
25	Выборка кристалла	—	*CS*	
26	Разрешение записи признаков	—	*WE*	Выход
27	Шина микрокоманды	0	*F*0	Вход
28		1	*F*1	
29		2	*F*2	
30	Не используется	—	—	—
31	Шина внешнего параметра сдвига	4	*SHI*4	Вход
32	Напряжение питания	—	*Ucc*	—
33	Шина внешнего параметра сдвига	3	*SHI*3	Вход
34		2	*SHI*2	
35		1	*SHI*1	
36		0	*SHI*0	
37	Выбор параметра сдвига	—	*SHS*	
38	Шина параметра сдвига	0	*SHB*0	Двунаправленный
39		1	*SHB*1	
40		2	*SHB*2	
41		3	*SHB*3	
42		4	*SHB*4	

Примечание. Выходы *D*15 ... *D*0, *SHB*4 ... *SHB*0 трехстабильные; выходы *F*, *OW* *ZR* с открытым коллектором.

Таблица 8.8. **Операции БИС АР**

Мнемоника	Код операции F0 F1 F2 S4	Операция	Результат операции: R15 ... RN	Результат операции: R (N — 1) ... R0 (N ≠ 0)
ПЛЕ	0 0 0 0/1	Поиск левой единицы	См. примечание	
СЛЛСР	0 0 1 0	Сдвиг логический влево со склейкой результата	D (15 — N) ∨ P15 ... D0 ∨ PN	P (N — 1) ... P0
	0 0 1 1		0	15 ... D (16 — N)
СЛПСР	0 1 0 0	Сдвиг логический вправо со склейкой результата	D (15 — N) ... D0	0
	0 1 0 1		P15 ... PN	D15 ∨ P (N — 1) ... D (16 — N) ∨ P0
САЛСР	0 1 1 0	Сдвиг арифметический влево со склейкой результата	D (15 — N) ∨ P15 ... D0 ∨ PN	P (N — 1) ... P0
	0 1 1 1		D15	D15 ... D (16 — N)
СЦЛ	1 0 0 0	Сдвиг циклический влево	D (15 — N) ... D0	D15 ... D (16 — N)
СЦП	1 0 0 1	Сдвиг циклический вправо	D (15 — N) ... D0	D15 ... D (16 — N)
СЛЛ	1 0 1 0	Сдвиг логический влево	D (15 — N) ... D0	0
СЛП	1 0 1 1	Сдвиг логический вправо	0	D15 ... D (16 — N)
СЛСР	1 1 0 0	Сдвиг со склейкой результата влево	D (15 — N) ∨ P15 ... D0 ∨ PN	P (N — 1) ... P0
СПСР	1 1 0 1	Сдвиг со склейкой результата вправо	P15 ... PN	D15 ∨ P (N — 1) ... D (16 — N) ∨ P0
САЛ	1 1 1 0	Сдвиг арифметический влево	D (15 — N) ... D0	0
САП	1 1 1 1	Сдвиг арифметический вправо	D15	D15 ... D (16 — N

Примечание. $2^4R4 + 2^3R^3 + 2^2R^2 + 2R1 + R0 = 15 - i$, $R15 \dots R5 = 0$

Информация в РР		Вырабатываемые признаки					
P15 … PN	P (N − 1) … P0 (N ≠ 0)	$\overline{F}$ N=0	$\overline{F}$ N ≠ 0	OW N = 0	OW N ≠ 0	ZR	WE
Содержимое РР не изменяется		$\bigvee_{i=0}^{15} Di$		D15		D15	1
0	D15 … D (16 − N)	1	$\overline{D(16-N)}$	0	$\bigvee_{i=0}^{N-1} D(15-i)$		1
D (15 − N)…D0	0	$\overline{D(15-N)}$		0			0
0	D15 … D (16 − N)	1	$\overline{D(16-N)}$	0	$\bigvee_{i=0}^{N-1} D(15-i)$		0
D (15 − N)…D0	0	$\overline{D(15-N)}$		0			1
0	D15 … D (16 − N)	$\overline{D15}$	$\overline{D(16-N)}$	$\bigvee_{i=0}^{N-1} D(15-N)\oplus D(15-i)$			1
D (15 − N)…D0	0	$\overline{D(15-N)}$		0			0
0	D15 … D (16 − N)	1	$\overline{D(16-N)}$	0	$\bigvee_{i=0}^{N-1} D(15-i)$	$\bigwedge_{i=0}^{15} \overline{Ri}$	1
D (15 − N)…D0	0	$\overline{D(15-N)}$		0			1
0	D15 … D (16 − N)	1	$\overline{D(16-N)}$	0	$\bigvee_{i=0}^{N-1} D(15-i)$		1
D (15 − N)…D0	0	$\overline{D(15-N)}$		0			1
0	D15 … D (16 − N)	1	$\overline{D(16-N)}$	0	$\bigvee_{i=0}^{N-1} D(15-i)$		1
D (15 − N)…D0	0	$\overline{D(15-N)}$		0			1
0	D15 … D (16 − N)	$\overline{D15}$	$\overline{D(16-N)}$	$\bigvee_{i=0}^{N-1} D(15-N)\oplus D(15-i)$			1
D (15 − N).. D	0	$\overline{D(15-N)}$		0			1

где i — вес первого слева разряда, содержащего 1 ($i = -1$, если D15 … D0 = 0).

Число разрядов, на которое нужно сдвинуть информацию, задается двоичным кодом $N = 2^3SHI3 + 2^2SHI2 + 2^1SHI1 + SHI0$ при $SHS = 0$ ($S4 = SHI4$ определяет направление сдвига), а при $SHS = 1$ $N = 2^3SHB3 + 2^2SHB2 + 2^1SHB1 + SHB0$ ($S4 = SHB4$). Если сдвиг происходит влево, то N задается в прямом коде ($S4 = 0$), если вправо — N задается в дополнительном коде ($S4 = 1$).

В графе «Информация в РР» показана информация, загруженная в PP с выхода сдвигателя. При сдвиге влево столбец $P15 \ldots PN$ указывает несдвинутую информацию в PP, т. е. в те разряды PP, которые не соответствуют выдвинутым разрядам сдвигателя, записываются нули; столбец $P(N-1) \ldots P0$ ($N \neq 0$) дает информацию в PP, представляющую выдвинутые разряды сдвигателя. При сдвиге вправо — наоборот. Если $N = 0$, то информацию в графах «Результат операции» и «Информация в РР» следует смотреть только в левых столбцах.

При разрядности информации, большей 16, сдвиг производится в несколько тактов. Например, рассмотрим сдвиг 32-разрядного операнда на 5 разрядов влево. В первом такте на АР подаются младшие 16 разрядов и выполняется операция СЛЛ. Во втором такте на АР подаются старшие 16 разрядов исходного операнда и выполняется операция СЛЛСР ($S4 = 0$) или СЛСР, в результате чего выдвинутые в первом такте и запомненные в РР разряды записываются на место выдвинутых разрядов старшего слова операнда.

Статические и динамические параметры АР приведены в табл. 8.9 и табл. 8.10 соответственно. На рис. 8.7 представлены временные диаграммы работы АР.

Рассмотрим функционирование БИС АР.

Операнд, над которым производится сдвиг, подается на двунаправленную шину D на положительном уровне синхросигнала CLK за время установки t_{SU} до отрицательного фронта синхроимпульса. Отрицательным фронтом CLK этот операнд записывается во входном регистре буфера БД. На положительном уровне CLK подаются также параметр сдвига (он может приходить на двунаправленную шину SHB или шину SHI), сигнал выбора параметра SHS, сигнал выбора кристалла $\overline{CS}$ и код микрокоманды за соответствующие времена установки до отрицательного фронта CLK.

Параметр сдвига отрицательным фронтом CLK записывается в регистр $РПС$. Операнд и параметр сдвига на входных шинах надо удерживать после отрицательного фронта CLK в течение соответствующих времен, необходимых для записи в регистры правильной информации. Код микрокоманды и сигнал $\overline{CS}$ удерживают до прихода следующего положительного уровня CLK, менять их можно не раньше времени удержания относительно положительного фронта CLK. Результат сдвига на выходе получаем через

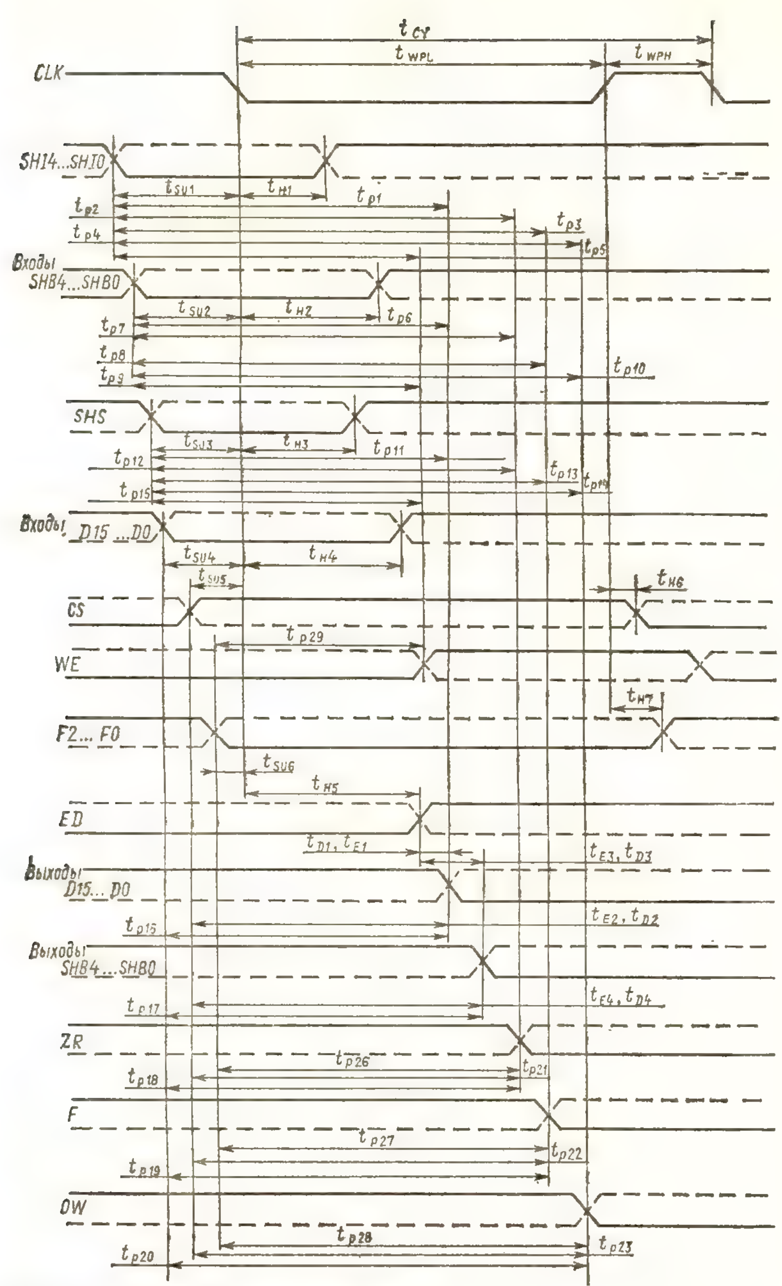


Рис. 8.7. Временна́я диаграмма работы БИС АР

Таблица 8.9. **Статические параметры микросхемы КР1802ВР1**

Параметр, единица измерения	Обозначение	Значение		Режим измерения
		мин.	макс.	
Входной ток «0», мА, для входов: *D*15 ... *D*0, *SHB*4 ... *SHB*0, *SHI*4 ... *SHI*0, *ED*, *F*2 ... *F*0 *CLK*, *SHS* *CS*	I_{IL}	 —0,4 —0,25 —0,8	 — — —	U_{CC} = 5,25 В U_{IL} = 0,45 В
Входной ток «1», мкА, для входов: *D*15 ... *D*0, *SHB*4 ... *SHB*0 *SHI*4 ... *SHI*0, *ED*, *F*2 ... *F*0 *CLK*, *CHS*, *CS*	I_{IH}	 — — —	 100 20 100	U_{CC} = 5,25 В U_{IH} = 5,25 В
Выходной ток «1», мкА, для выходов *ZR*, *OW*, *F*	I_{OH}	—	250	U_{CC} = 5,25 В U_{TL} = 0,8 В U_{TH} = 2,0 В
Выходное напряжение «0», В, для выходов: *D*15 ... *D*0, *SHB*4 ... *SHB*0 *ZR*, *OW*, *F*, *WE*	U_{OL}	 — —	 0,5 0,5	U_{CC} = 4,75 В U_{TL} = 0,8 В U_{TH} = 2,0 В I_{OL} = 15 мА I_{OL} = 10 мА
Выходное напряжение «1», В, для выходов *WE*, *D*15 ... *D*0, *SHB*4 ... *SHB*0	U_{OH}	2,4	—	U_{CC} = 4,75 В U_{TL} = 0,8 В U_{TH} = 2,0 В I_{OH} = —1 мА
Ток выключенного состояния, мкА, по выходам *D*15 ... *D*0, *SHB*4 ... *SHB*0	I_{OZ}	—	100	U_{CC} = 5,25 В U_{OH} = 5,25 В U_{TH} = 2 В U_{TL} = 0,8 В
Ток короткого замыкания, мА, на выходе	I_{OS}	—60	—15	U_{CC} = 5,0 В U_{OL} = 0 В
Ток потребления, мА	I_{CC}	—	280	U_{CC} = 5,25 В U_{IH} = 2,4 ... 4,5 В для входа *CLK*

Таблица 8.10. **Динамические параметры микросхемы КР1802ВР1**

Параметр	Обозначение	Значения, нс		
		мин.	ном.	макс.
Длительность цикла	t_{CY}	210	150	—
Длительность положительного импульса	t_{WPH}	70	50	—
Длительность отрицательного импульса	t_{WPL}	70	50	—
Время установки сигнала относительно отрицательного фронта синхроимпульса для входов:				
*SHI*4 ... *SHI*0	t_{SU1}	25	15	—
*SHB*4 ... *SHB*0	t_{SU2}	20	10	—
SHS	t_{SU3}	35	25	—
*D*15 ... *D*0	t_{SU4}	10	10	—
CS	t_{SU5}	10	0	—
*F*2 ... *F*0	t_{SU6}	40	30	—
Время удержания сигнала относительно отрицательного фронта синхроимпульса для входов:				
*SHI*4 ... *SHI*0	t_{H1}	25	15	—
*SHB*4 ... *SHB*0	t_{H2}	30	20	—
SHS	t_{H3}	30	20	—
*D*15 ... *D*0	t_{H4}	30	20	—
ED	t_{H5}	20	15	—
Время удержания сигнала, относительно положительного фронта синхроимпульса для входов:				
CS	t_{H6}	30	10	—
*F*2 ... *F*0	t_{H7}	30	10	—
Время задержки распространения сигнала от входов *SHI*4 ... *SHI*0 до выходов:				
*D*15 ... *D*0	$t_{\text{р}1}$	—	120	170
ZR	$t_{\text{р}2}$	—	125	175
F	$t_{\text{р}3}$	—	105	150
OW	$t_{\text{р}4}$	—	105	150
WE	$t_{\text{р}5}$	—	40	50
То же от входов *SHB*4 ... *SHB*0 до выходов:				
*D*15 ... *D*0	$t_{\text{р}6}$	—	115	160
ZR	$t_{\text{р}7}$	—	120	168
F	$t_{\text{р}8}$	—	45	80
WE	$t_{\text{р}9}$	—	40	45
OW	$t_{\text{р}10}$	—	100	150
То же от входа *SHS* до выходов:				
*D*15 ... *D*0	$t_{\text{р}11}$	—	130	190
ZR	$t_{\text{р}12}$	—	135	190
F	$t_{\text{р}13}$	—	60	84
OW	$t_{\text{р}14}$	—	115	160
WE	$t_{\text{р}15}$	—	50	75
То же от входов *D*15 ... *D*0 до выходов:				
*D*15 ... *D*0	$t_{\text{р}16}$	—	115	160
*SHB*4 ... *SHB*0	$t_{\text{р}17}$	—	115	170
ZR	$t_{\text{р}18}$	—	120	180
F	$t_{\text{р}19}$	—	105	147
OW	$t_{\text{р}20}$	—	120	170

Параметр	Обозначение	Значения, нс		
		мин.	ном.	макс.
То же от входа CS до выходов:				
ZR	t_{p21}	—	40	56
F	t_{p22}	—	40	56
OW	t_{p23}	—	40	56
То же от входов $F2 \dots F0$ до выходов:				
$D15 \dots D0$	t_{p24}	—	125	180
$SHB4 \dots SHB0$	t_{p25}	—	125	180
ZR	t_{p26}	—	130	160
F	t_{p27}	—	120	168
OW	t_{p28}	—	135	160
WE	t_{p29}	—	30	45
Время отключения сигнала от входа ED до выходов:				
$D15 \dots D0$	t_{D1}	—	25	63
$SHB4 \dots SHB0$	t_{D3}	—	40	56
Время включения сигнала от входа ED до выходов:				
$D15 \dots D0$	t_{E1}	—	45	73
$SHB4 \dots SHB0$	t_{E3}	—	60	84
Время отключения сигнала от входа CS до выходов:				
$D15 \dots D0$	t_{D2}	—	25	35
$SHB4 \dots SHB0$	t_{D4}	—	40	56
Время включения сигнала от входа CS до выходов:				
$D15 \dots D0$	t_{E2}	—	45	63
$SHB4 \dots SHB0$	t_{E4}	—	60	84

время задержки распространения, предварительно подав сигнал $\overline{ED}$ за время t_{E1} до формирования достоверного результата сдвига. Выдвинутые разряды по положительному фронту CLK (при $\overline{CS}$ = «0») записываются в РР.

Через соответствующие времена распространения вырабатываются признаки $\overline{F}$, OW, ZR, WE, которые используются для анализа результатов сдвига. Эти признаки формируются при выбранном кристалле $\overline{CS}$ = «0». Если кристалл не выбран, они равны «1».

8.4. РЕГИСТРЫ ОБЩЕГО НАЗНАЧЕНИЯ КР1802ИР1

БИС регистров общего назначения (РОН) предназначена для построения сверхоперативных запоминающих устройств (СОЗУ), т. е. ЗУ небольшой емкости, но с высоким быстродействием (время цикла порядка 100 нс). На основе БИС РОН могут строиться буферные ЗУ при обмене информацией между отдельными устрой-

ствами, ЗУ команд, в которые осуществляется «закачка» команд из большой памяти. БИС РОН можно использовать в качестве отдельных регистров в различных устройствах обработки информации.

Условное графическое обозначение РОН приведено на рис. 8.8, назначение выводов — в табл. 8.11.

Структурная схема БИС РОН показана на рис. 8.9.

БИС РОН имеет два дешифратора *ДШ*1 и *ДШ*2 на 4 входа и 16 выходов каждый, *накопитель*, представляющий собой матрицу 16 × 4 на статических триггерах (запись в них осуществляется уровнем и не зависит от параметров фронта), 8 усилителей считывания *УС* с трехстабильными выходами, 8 усилителей записи *УЗ*.

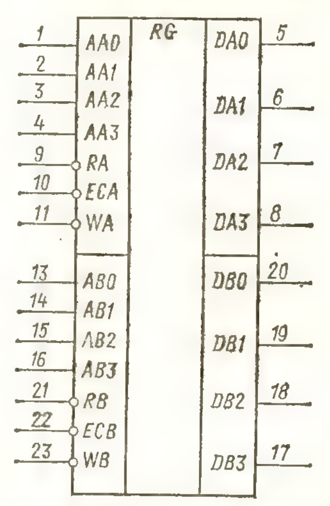


Рис. 8.8. Условное графическое обозначение БИС РОН

Режимы работы БИС РОН представлены в табл. 8.12. Буквой X в таблице обозначено безразличное состояние сигнала. Если на входах *ECA* и *ECB* «1», то выходы каналов *A* и *B* будут в состоянии высокого сопротивления (третье состояние) и независимо от сигналов $\overline{RA}$, $\overline{WA}$, $\overline{RB}$ и $\overline{WB}$ запись в накопитель или считывание из него информации не происходит. В табл. 8.12 это обозначено *NOP* (нет операции).

При подаче «0» на вход *ECA* (*ECB*) разрешается работа по данному каналу и режим работы определяется комбинацией сигналов $\overline{RA}$, $\overline{WA}$ ($\overline{RB}$, $\overline{WB}$). Подача «0» одновременно на *RA* и *WA* (*RB* и *WB*) запрещается, поскольку это состояние не определено и информация в выбранном регистре непредсказуема.

При $\overline{RA}$ = «0» происходит чтение информации в канал *A*, при $\overline{RB}$ = «0» в канал *B*. При одинаковых кодах адреса можно читать один и тот же регистр и в канале *A* и в канале *B*. При $\overline{WA}$ = = «0» ($\overline{WB}$ = «0») идет запись информации с канала *A* (*B*) в соответствующий регистр. Следует помнить, что информацию с канала *A* можно записать по адресу, подаваемому только на входы адреса *AA*3 ... *AA*0, а с канала *B* — только на входы адреса *AB*3 ... *AB*0. То же самое относится и к чтению в канал *A* и *B*.

При $\overline{WA} = 0$ и $\overline{WB} = 0$ происходит запись одновременно с каналов *A* и *B* в регистры накопителя, причем адреса каналов должны быть разные. Если адреса одинаковые, то результат записи не определен.

Рис. 8.9. Структурная схема БИС РОН

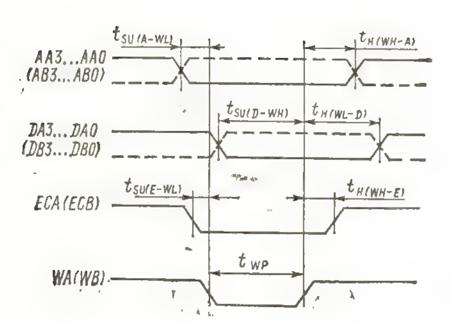


Рис. 8.10. Временная диаграмма работы РОН при записи

При записи с канала A ($\overline{WA}$ = «0») и считывании в канал B (или наоборот) адреса каналов A и B могут быть одинаковыми.

Для примера рассмотрим временную диаграмму для записи информации в РОН. Чтобы записать информацию, ее надо подать на канал A (B) за время установки до положительного фронта импульса записи. Адрес, по которому должна записаться информация, подается за время установки до отрицательного фронта импульса записи $t_{SU(A-WL)}$. На вход разрешения обмена по каналу A (B) за время установки $t_{SU(E-WL)}$ до отрицательного импульса фронта записи надо подать «0». Длительность самого импульса записи должна быть не меньше t_{WP}, указанной в таблице динамических параметров. После положительного фронта импульса записи надо удерживать адрес, данные и сигнал разрешения обмена на время $t_{H(WH-A)}$, $t_{H(WH-D)}$ и $t_{H(WH-E)}$ соответственно.

Таблица 8.11. **Назначение выводов микросхемы КР1802ИР1**

Номер вывода	Назначение	Разряд	Обозначение	Тип
1	Шина адреса канала *A*	0	*AA*0	Вход
2		1	*AA*1	
3		2	*AA*2	
4		3	*AA*3	
5	Шина данных канала *A*	0	*DA*0	Двунаправленный
6		1	*DA*1	
7		2	*DA*2	
8		3	*DA*3	
9	Сигнал считывания информации канала *A*	—	*RA*	Вход
10	Разрешение обмена с каналом *A*	—	*ECA*	
11	Сигнал записи информации с канала *A*	—	*WA*	
12	Общий	—	*GND*	—
13	Шина адреса канала *B*	0	*AB*0	Вход
14		1	*AB*1	
15		2	*AB*2	
16		3	*AB*3	
17	Шина данных канала *B*	3	*DB*3	Двунаправленный
18		2	*DB*2	
19		1	*DB*1	
20		0	*DB*0	
21	Сигнал считывания информации канала *B*	—	*RB*	Вход
22	Разрешение обмена с каналом *B*	—	*ECB*	
23	Сигнал записи информации с канала *B*	—	*WB*	
24	Напряжение питания	—	U_{CC}	—

Примечание. Выходы *DA*3 ... *DA*0, *DB*3 ... *DB*0 трехстабильные.

Таблица 8.12. Режим работы микросхемы КР1802ИР1

Канал A					Канал B					Режим работы
$\overline{ECA}$	$\overline{RA}$	$\overline{WA}$	AA3 ... AA0	Операция	$\overline{ECB}$	$\overline{RB}$	$\overline{WB}$	$AB3$ ... $AB0$	Операция	
1 X	X 1	X 1	X X	NOP NOP	1 X	X 1	X 1	X X	NOP NOP	Выключенное (третье состояние) на выходах $DA3 \dots DA0$, $DB3 \dots DB0$ канала A и B соответственно
1	X	X	X	NOP	0	0	1	K	$(RK) \to DB$	Считывание содержимого регистра RK в канал B
1	X	X	X	NOP	0	1	0	K	$DB \to (RK)$	Запись информации с канала B в регистр RK
0	0	1	N	$(RN) \to DA$	1	X	X	X	NOP	Считывание содержимого регистра RN в канал A
0	1	0	N	$DA \to (RN)$	1	X	X	X	NOP	Запись информации с канала A в регистр RN
0	0	1	N	$(RN) \to DA$	0	0	1	K	$(RK) \to DB$	Считывание содержимого регистров RN и RK в каналы A и B соответственно
0	0	1	N	$(RN) \to DA$	0	1	0	K	$DB \to (RK)$	Считывание содержимого RN в канал A и запись информации с канала B в регистр RK
0	1	0	N	$DA \to (RN)$	0	0	1	K	$(RK) \to DB$	Запись информации с канала A в регистр RN и считывание содержимого регистра RK в канал B
0	1	0	N	$DA \to (RN)$	0	1	0	K	$DB \to (RK)$	Запись информации с каналов A и B в регистры RN и RK

П р и м е ч а н и е. N и K могут принимать значения 0 ... 15.

Таблица 8.13. **Статические параметры микросхемы КР1802ИР1**

Параметр, единица измерения	Обозначение	Значения параметров		Режим измерения
		мин.	макс.	
Входной ток «0», мА, для входов:	I_{IL}			$U_{CC} = 5{,}25$ В $U_{IL} = 0{,}45$ В
*DA*3 ... *DA*0, *DB*3 ... *DB*0			—0,25	
RA, *RB*, *WA*, *WB*			—0,4	
*AA*3 ... *AA*0, *AB*3 ... *AB*0			—0,25	
ECA, *ECB*			—0,8	
Входной ток «1», мкА, для входов:	I_{IH}			$U_{CC} = 5{,}25$ В $U_{IH} = 5{,}25$ В
RA, *RB*, *WA*, *WB*, *AA*3 ... *AA*0, *AB*3 ... *AB*0		—	20	
ECA, *ECB*		—	40	
Входное напряжение, В: «1»	U_{IH}	2,0	5,25	—
«0»	U_{IL}	0	0,8	
Выходное напряжение, В: «1»	U_{OH}	2,4	—	$U_{CC} = 4{,}75$ В $I_{OH} = -1{,}6$ мА $I_{OL} = 15$ мА
«0»	U_{OL}	—	0,5	
Ток выключенного состояния, мкА	I_{OZ}	—	100	$U_{CC} = 5{,}25$ В $U_{OH} = 5{,}25$ В $U_{TH} = 2{,}0$ В $U_{TL} = 0{,}8$ В
Ток короткого замыкания на выходе, мА	I_{OS}	—65	—15	$U_{CC} = 5{,}0$ В $U_{OL} = 0$ В
Ток потребления, мА	I_{CC}	—	180	$U_{CC} = 5{,}25$ В $U_{IH} = 4{,}5$ В

Таблица 8.14. **Динамические параметры микросхемы КР1802ИР1**

Параметр	Обозначение	Значения, нс		
		мин.	ном.	макс.
Время выбора микросхемы	t_{CS}	—	25	37
Длительность импульса записи	t_{WP}	37	25	—
Время установки адреса на входах *AA*3 ... *AA*0, *AB*3 ... *AB*0 относительно отрицательного фронта импульса записи на входах *WA*, *WB*	$t_{SU\,(A-WL)}$	15	7	—

Продолжение табл. 8.14

Параметр	Обозначение	Значение, нс		
		мин.	ном.	макс.
Время установки информации на входах $DA3 \dots DA0$, $DB3 \dots DB0$ относительно положительного фронта импульса записи WA, WB	$t_{SU\ (D-WH)}$	37	25	—
Время установки сигналов ECA, ECB относительно отрицательного фронта импульса записи на входах WA, WB	$t_{SU\ (E-WL)}$	10	5	—
Время удержания адреса на входах $AA3 \dots AA0$, $AB3 \dots AB0$ относительно положительного фронта импульса записи на входах WA, WB	$t_{H\ (WH-A)}$	10	5	—
Время удержания сигналов ECA, ECB относительно положительного фронта импульса записи на входах WA, WB	$t_{H\ (WH-E)}$	10	5	—
Время удержания информации на входах $DA3 \dots DA0$, $DB3 \dots DB0$ относительно положительного фронта импульса записи на входах WA, WB	$t_{H\ (WH-D)}$	10	5	—
Время считывания от адреса	$t_{p\ (A-D)}$	—	50	60
Время записи-чтения	$t_{p\ (WL-D)}$	—	43	73
Время передачи информации	$t_{p\ (D-D)}$	—	40	68
Время считывания от сигналов чтения	t_R	—	25	37
Время выключения выходов $DA3 \dots DA0$, $DB3 \dots DB0$ от входов:				
RA, RB	$t_{D\ (R-D)}$	—	25	37
ECA, ECB	$t_{D\ (E-D)}$	—	25	37

Статические и динамические параметры микросхемы КР1802ИР1 приведены в табл. 8.13 и 8.14 соответственно. На рис. 8.10—8.12 представлены временные диаграммы работы РОН.

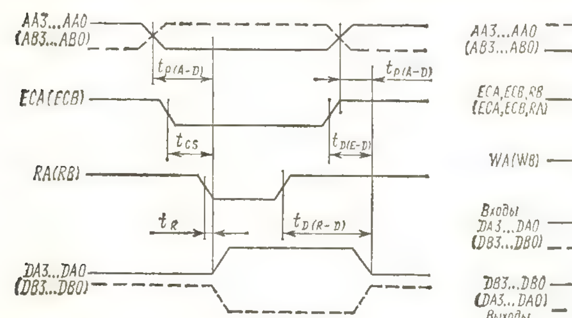


Рис. 8.11. Временна́я диаграмма работы РОН при считывании

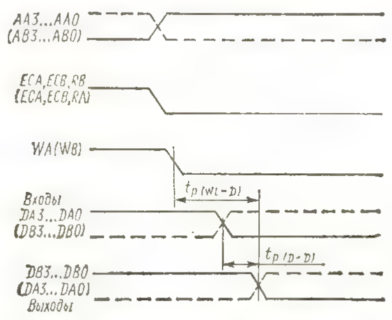


Рис. 8.12. Временна́я диаграмма работы РОН в режиме «Запись—чтение»

Для построения буферных ЗУ емкости бо́льшей, чем 16 × 4, увеличивают число микросхем, наращивая РОН как в сторону увеличения разрядов, так и в сторону увеличения слов.

8.5. МИКРОСХЕМА УМНОЖИТЕЛЯ КР1802ВР2

Микросхема умножителя (УМН) предназначена для аппаратной реализации операций умножения и деления двоичных кодов и чисел, представленных в дополнительном коде. Операция умножения производится над двумя 8-разрядными операндами, результат — 16-разрядный. Операция деления производится над 16-разрядным делимым и 8-разрядным делителем с получением результата в виде 8 разрядов частного и 8 разрядов остатка.

Для получения устройств большей разрядности (кратной восьми) используют несколько БИС. Наличие вывода, определяющего старший кристалл (*CHS*), и выводов выборки кристалла (*CS*) позволяет при наращивании производить операции без введения дополнительных связей с выработкой признаков результата. Прием исходных данных и выдача результата производятся по одним и тем же двунаправленным шинам.

Рис. 8.13. Условное графическое обозначение БИС УМН

Условное графическое обозначение УМН представлено на рис. 8.13. Назначение выводов УМН описано в табл. 8.15.

Структурная схема БИС УМН (рис. 8.14) включает следующие устройства:

регистр инструкции *РИ* (двухразрядный); обеспечивает хранение кода операции во время ее выполнения. Запись кода операции осуществляется при $\overline{CS1}$ = «0»;

регистры *Р1*, *Р2*, *Р3*; *Р1* (8-разрядный) служит для записи множителя в операциях умножения и младших восьми разрядов делимого при делении. В процессе выполнения операций умножения и деления в *Р1* записываются соответственно младшие разряды произведения и частное. Состоит из двух однотактных регистров *Р1′* и *Р1″*. *Р2* (8-разрядный однотактный) служит для хранения множимого и делителя в операциях умножения и деления соответственно. *Р3* (10-разрядный двухтактный) предназначен для записи старших восьми разрядов делимого при делении. В *Р3* записывается очередное частичное произведение в процессе выполнения операции умножения и очередной остаток приделении. После выполнения операции в *Р3* содержатся

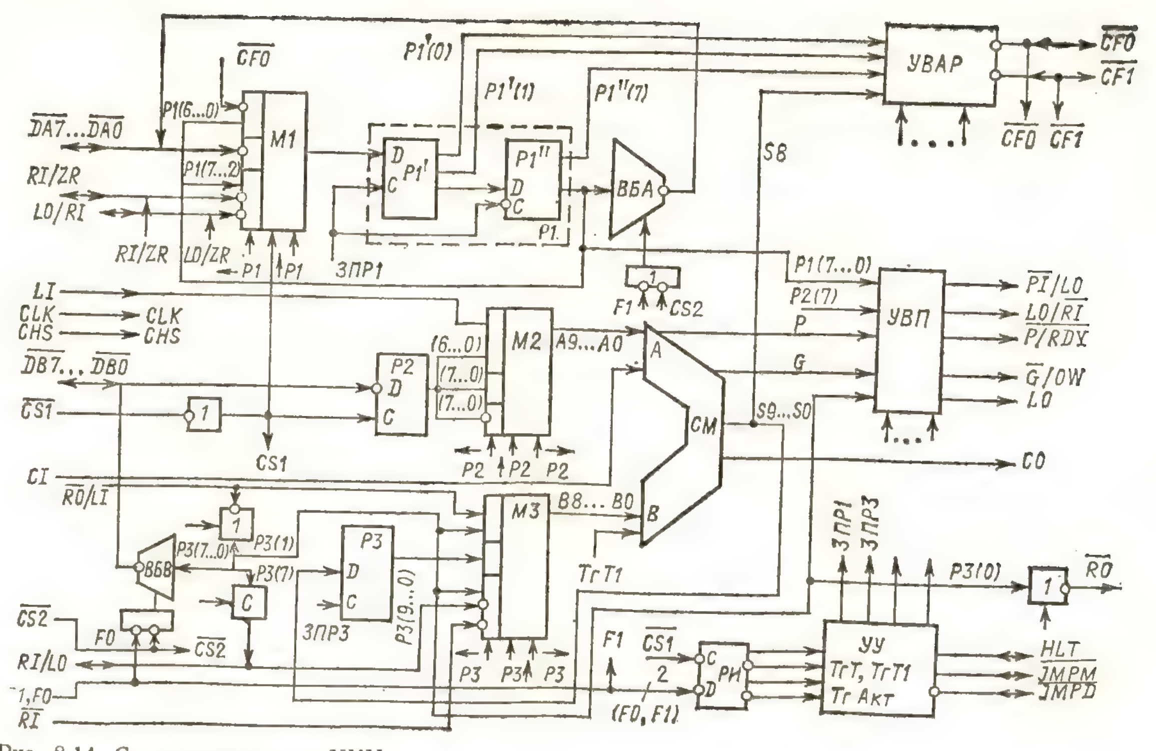


Рис. 8.14. Структурная схема УМН

Таблица 8.15. **Назначение выводов микросхемы КР1802ВР2**

Номер вывода	Назначение	Разряд	Обозначение	Тип
1	Шина данных *B*	6	*DB*6	Двунаправленный
2		7	*DB*7	
3	7-й разряд Р3	—	*RI*/*LO*	
4	Общий	—	*GND*	—
5	Выход для запуска счетчика циклов при умножении, вход для запуска счетчика циклов при делении	—	*JMPM*	Двунаправленный
6	Выход для запуска счетчика циклов при делении, вход для запуска счетчика циклов при умножении	—	*JMPD*	
7	7-й разряд Р2, признак «Знак», выдача «1» или «0»	—	*LO*	Выход
8	Сигнал «Останов»	—	*HLT*	Двунаправленный
9	Выбор микросхемы 1	—	*CS*1	Вход
10	Шина микроинструкции	1	*F*1	
11	Перенос	—	*CO*	Выход
12	Выбор старшего кристалла	—	*CHS*	Вход
13	Шина микроинструкции	0	*F*0	
14	Синхронизация	—	*CLK*	
15	Выбор микросхемы 2	—	*CS*2	
16	Вход в 6-й разряд Р1, выход признака *ZR*	—	*RI*/*ZR*	Двунаправленный
17	Шина управления микрооперациями	1	*CF*1	
18	Вход-выход 7-го разряда Р1, выход признака «Расширение»	—	*LO*/*RI*	
19	Общий	—	*GND*	—
20	Шина управления микрооперациями	0	*CF*0	Двунаправленный
21	Шина данных *A*	7	*DA*7	
22		6	*DA*6	
23		5	*DA*5	
24		4	*DA*4	
25		3	*DA*3	
26		2	*DA*2	
27		1	*DA*1	
28		0	*DA*0	
29	Перенос	—	*CI*	Вход
30	Выход первого, вход в нулевой разряд Р3	—	*RO*/*LI*	Двунаправленный
31	Нулевой разряд Р3	—	*R*0	Выход
32	Напряжение питания	—	U_{CC}	—
33	Сигнал распространения переноса, сигнал признака «Готов»	—	*P*/*RDY*	Выход
34	Сигнал генерации переноса, сигнал переполнения	—	*G*/*OW*	
35	6-й разряд Р3	—	*RI*	Вход
36	Нулевой разряд Р2	—	*LI*	

Номер вывода	Назначение	Разряд	Обозначение	Тип
37 38 39 40 41 42	Шина данных *B*	0 1 2 3 4 5	*DB*0 *DB*1 *DB*2 *DB*3 *DB*4 *DB*5	Двунаправленный
Примечание. Выходы *DA*7 ... *DA*0, *DB*7 ... *DB*0, *RI*/*LO*, *JMPM*, *JMPD*, *RO*/*LI*, *RO*, *CF*1, *CF*0 трехстабильные, выходы *HLT*, *RI*/*ZR*, *LO*/*RI* с открытым коллектором.				

соответственно старшие разряды произведения при умножении и остаток при делении;

мультиплексоры *M1*, *M2*, *M3*; *M1* обеспечивают прием операнда по магистрали и запись его в *P1* в операциях умножения и деления, а также сдвиг содержимого *P1* на два разряда вправо (в сторону младших разрядов) при умножении и на один разряд влево (в сторону старших разрядов) при делении. *M2* пропускает на вход *A* сумматора *CM* содержимое *P2*, содержимое *P2* со сдвигом влево на один разряд (в операциях умножения), инвертированное содержимое *P2* и код нуля при выполнении операций умножения и деления. *M3* пропускает на вход *B* сумматора *CM* содержимое *P3* со сдвигом вправо на два разряда (в операциях деления) и код нуля;

устройство управления *УУ*; предназначено для генерации управляющих сигналов, обеспечивающих выполнение всех действий заданных кодом операции;

сумматор *CM*; служит для вычисления частичных произведений и коррекции результата при умножении, вычисления остатков для получения очередного бита частного при делении и восстановления окончательного остатка;

выходной буфер выдачи результата по магистрали *A ВБА* и *B ВБВ*; служит для выдачи информации из регистров *P1* и *P3* на магистрали *DA* и *DB* соответственно;

устройство выдачи признаков *УВП*; предназначено для выдачи выдвигаемых разрядов, различных признаков результата и сигналов ускоренного переноса;

устройство выдачи анализируемых признаков *УВАР*; служит для выдачи анализируемых разрядов при умножении и делении в БИС УМН, которая в данный момент активна, и приема этих разрядов всеми остальными схемами.

Микросхема УМН выполняет следующие операции: умножение 8-разрядных целых чисел, представленных в дополнительном

коде; умножение 8-разрядных кодов; деление кодов; загрузку старшего слова делимого; чтение результата.

Все операции (кроме чтения) инициируются сигналом $\overline{CS1}$ и определяются двухразрядным кодом, заданным сигналами на выводах $F0$ и $F1$. Выполнение этих операций синхронизируется синхроимпульсами CLK.

Чтение результата инициируется сигналом $\overline{CS2}$.

Список операций с указанием выполняемых действий и правил формирования признаков результата даны в табл. 8.16.

Операция «Загрузка» предназначена для записи в регистр Р3 старшего слова делимого.

В связи с тем, что в УМН имеется два приемных регистра и предусмотрена работа с двумя магистралями, то одновременно с кодом операции БИС УМН может принять два 8-разрядных операнда. Так как используется 16-разрядное делимое, то старшее слово делимого загружается предварительно по операции «Загрузка».

По сигналу $\overline{CS1}$ устройство управления переводит БИС в состояние «Не готова» ($\overline{P/RDY}$ = «1»), записывает код операции в РИ, разрешает прием операнда с магистрали DB в регистр Р2. После снятия сигнала $\overline{CS1}$ содержимое регистра Р2 подключается ко входу A сумматора, на вход B сумматора подается код нуля. Результат суммирования записывается в Р3, после чего УУ переводит БИС в состояние «Готова» ($\overline{P/RDY}$ = «0»). По окончании операции «Загрузка» признаки результата не формируются.

Операции «Умножение чисел» и «Умножение кодов» производятся следующим образом.

По сигналу $\overline{CS1}$ УУ переводит БИС УМН в состояние «Не готова», записывает код операции в РИ и разрешает запись множителя и множимого с магистралей DA и DB в регистры Р1 и Р2 соответственно. После снятия сигнала $\overline{CS1}$ начинает выполняться операция.

В УМН реализован циклический алгоритм умножения с логическим ускорением за счет группировки разрядов множителя по парам, анализа пары, на которую производится умножение множимого в данном цикле и накоплением суммы частичных произведений.

Поскольку числа представлены в дополнительном коде, в алгоритме предусмотрена однотактная коррекция результата умножения. Различие алгоритмов умножения кодов и чисел состоит только в разнице коррекции.

При умножении младшие разряды регистра Р3 (0, 1) постоянно подаются на выводы $R0$ и RO/LI.

Анализируемая пара разрядов множителя подается на выводы $CF0$, $CF1$.

Таблица 8.16. **Операции БИС УМН**

Мнемоника		Код операции	Операция	Выполнение действия	Признаки результата (после окончания операции)				
латинская	русская	$F0\ F1\ \overline{CS1}\ \overline{CS2}$			переполнение	расширение	нуль	знак	знак
					$\overline{G}/OW$	$LO/\overline{RI}$	$\overline{RI}/ZR$	LO	$\overline{CF1}$
MPI	УМЧ	0 0 0 1	Умножение чисел	$DA \rightarrow P1$ $DB \rightarrow P2$ $P3,\ P1 := = P1 \times P2$	0	1 при $P3(i) = \overline{CF1}$, $i = 0 \ldots 7$ 0 Не равно	$\bigwedge_{i=0}^{7} \overline{P1}(i) \wedge \wedge \overline{P3}(i)$	$P2(7) \wedge \overline{CHS} \vee \wedge P3(7)$	$\overline{P1\ (7)}$
MPC	УМК	0 1 0 1	Умножение кодов		0	$\bigwedge_{i=0}^{7} P3\ (i)$		$P2(7) \wedge \overline{CHS}$	$\overline{P1\ (7)}$
DVC	ДЛК	1 0 0 1	Деление кодов	$DA \rightarrow P1$ $DB \rightarrow P2$ $P1,\ P3 := = P3,\ P1 : P2$	$\overline{TrT}$	1	$\bigwedge_{i=0}^{7} \overline{P1}\ (i)$	$P2\ (7) \wedge \overline{CHS}$	$\overline{P1\ (7)}$
LDR	ЗГР	1 1 0 0	Загрузка	$DB \rightarrow P3$	0	1	0	$P2\ (7) \wedge \overline{CHS}$	$\overline{P1\ (7)}$
RDL	ЧР1	1 0 1 0	Чтение Р1	$P1 \rightarrow DA$	Выдаются признаки, выработанные в последней операции не чтения (УМЧ, УМК, ДЛК, ЗГР)				
RDH	ЧР3	0 1 1 0	Чтение Р3	$P3 \rightarrow DB$					
RDW	ЧГД	0 0 1 0	Чтение Р1 и Р3	$P3 \rightarrow DB$ $P1 \rightarrow DA$					

Значения разрядов Р1 (0) и Р1 (1) выдаются на выводы *CF0* и *CF1* в обратном коде. Обозначим прямой код содержимого Р1 (0) через У2, а прямой код содержимого Р1 (1) через У1.

В зависимости от значений У1, У2 и триггера переноса (ТгТ), который первоначально обнуляется, а в дальнейшем устанавливается согласно значениям переноса, в УМН выполняются следующие действия:

1. У1 = 0; У2 = 0; ТгТ = 0.

При этом множимое умножается на число 00 и происходит умножение на 4 (сдвиг вправо на два разряда) суммы предыдущих частичных произведений. Для этого на вход *A* сумматора подается «0», на входе *B* — сдвинутое на два разряда вправо содержимое Р3 (на первом шаге на вход *B* подается «0»), в котором хранилась предыдущая сумма. В старшие разряды входа *B* подается знак предыдущей части произведения, хранящийся на специальном триггере ТгТ1, если УМН старшая, а при *CHS* = «0» информация с выводов *RI* и *RI/LO*, т. е. выдвигаемые младшие разряды более старшей БИС. Новая сумма частичных произведений записывается в Р3.

В триггер ТгТ записывается «0», так как отсутствует перенос в следующую пару множителя (ТгТ = 0).

2. У1 = 0; У2 = 0; ТгТ = 1.

Здесь множимое фактически необходимо умножить на число 01. Поэтому на вход *A* сумматора подается прямой код Р2. В старшие разряды *A* подается знак множимого (в случае умножения кодов, когда 7-й знаковый разряд числа занят значащей цифрой на А9, А8, подается «0»).

Остальные действия аналогичны п. 1.

3. У1 = 0; У2 = 1; ТгТ = 0.

Поскольку умножение производится также на 01, все действия аналогичны п. 2.

4. У1 = 0; У2 = 1; ТгТ = 1.

Поскольку здесь присутствует сигнал переноса из младшей пары множителя, умножение фактически производится на число 10, поэтому на вход *A* сумматора подается прямой код Р2, сдвинутый на один разряд влево. В младший разряд *A* входа сумматора подается информация с вывода *LI*, т. е. старший разряд более младшей схемы или «0», если таковая отсутствует.

Остальные действия аналогичны п. 1.

5. У1 = 1; У2 = 0; ТгТ = 0.

Так как умножение происходит на число 10, то действия полностью совпадают с п. 4.

6. У1 = 1; У2 = 0; ТгТ = 1.

При этом множимое необходимо умножить на число 11. Чтобы не делать двух тактов суммы-сдвига, используется искусственный прием, когда число 11 заменяется 100-1, т. е. в этом цикле производится операция вычитания множимого и формируется единица переноса в следующую пару множителя. Поэтому на вход *A*

сумматора подается обратный код Р2 плюс 1 для образования дополнительного кода от кода, принятого на Р2. Так как при этом необходимо передать единицу переноса в следующую пару множителя, то ТгТ = 1. Остальные действия аналогичны п. 1.

7. У1 = 1; У2 = 1; ТгТ = 0.

Действия аналогичны п. 6.

8. У1 = 1; У2 = 1; ТгТ = 1.

Множимое необходимо умножить на число 100, поэтому можно считать, что умножение производится на число 00 и существует перенос в следующую пару множителя, т. е. действия аналогичны п. 1 за исключением того, что ТгТ = 1.

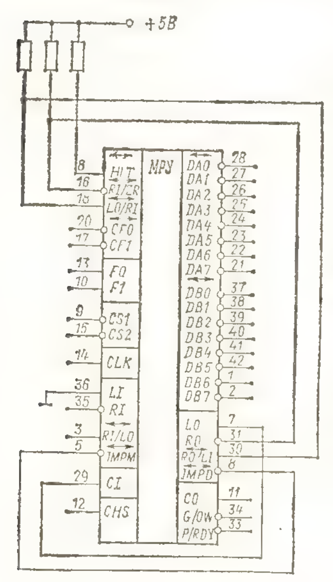


Рис. 8.15. Соединение выводов одной схемы УМН

Из описания следует, что получаемое произведение записывается в Р3. Однако, так как регистр Р3 10-разрядный, а должен хранить по меньшей мере 16 разрядов, в процессе умножения младшие разряды произведения из регистра Р3 передаются в регистр Р1 по два в каждом цикле. Разряды Р1 освобождаются от очередной пары множителя после проведения соответствующего цикла умножения.

Отдельные действия, производимые в процессе умножение (занесение младших разрядов в Р1, получение дополнительного кода Р2), выполняются соответствующими соединениями выводов УМН (рис. 8.15).

При соединении нескольких УМН (пример соединения двух БИС дан на рис. 8.16, правила объединения БИС — в табл. 8.17) умножение происходит аналогично, т. е. регистры и сумматоры нескольких УМН являются расширенными регистрами и сумматорами.

Старшая БИС УМН при этом выбирается сигналом CHS = «1». Эта БИС в нужные моменты времени выдает на вывод LO единицу, которая передается на вход CI самой младшей БИС УМН для получения дополнительного кода (пп. 6, 7), однако замена разрядов множителя разрядами произведения происходит только в той БИС, которая в этот момент является активной, т. е. пара множителей которой в этот момент анализируется.

Активность УМН характеризуется установкой специального триггера (ТгАкт) в 1 (см. рис. 8.14) по сигналу $JMPD$, который поступает на вывод $JMPD$ данной БИС с вывода $JMPM$ более младшей БИС.

Через четыре цикла сложения-сдвига, после которых часть множителя, записанная в данной БИС УМН, полностью про-

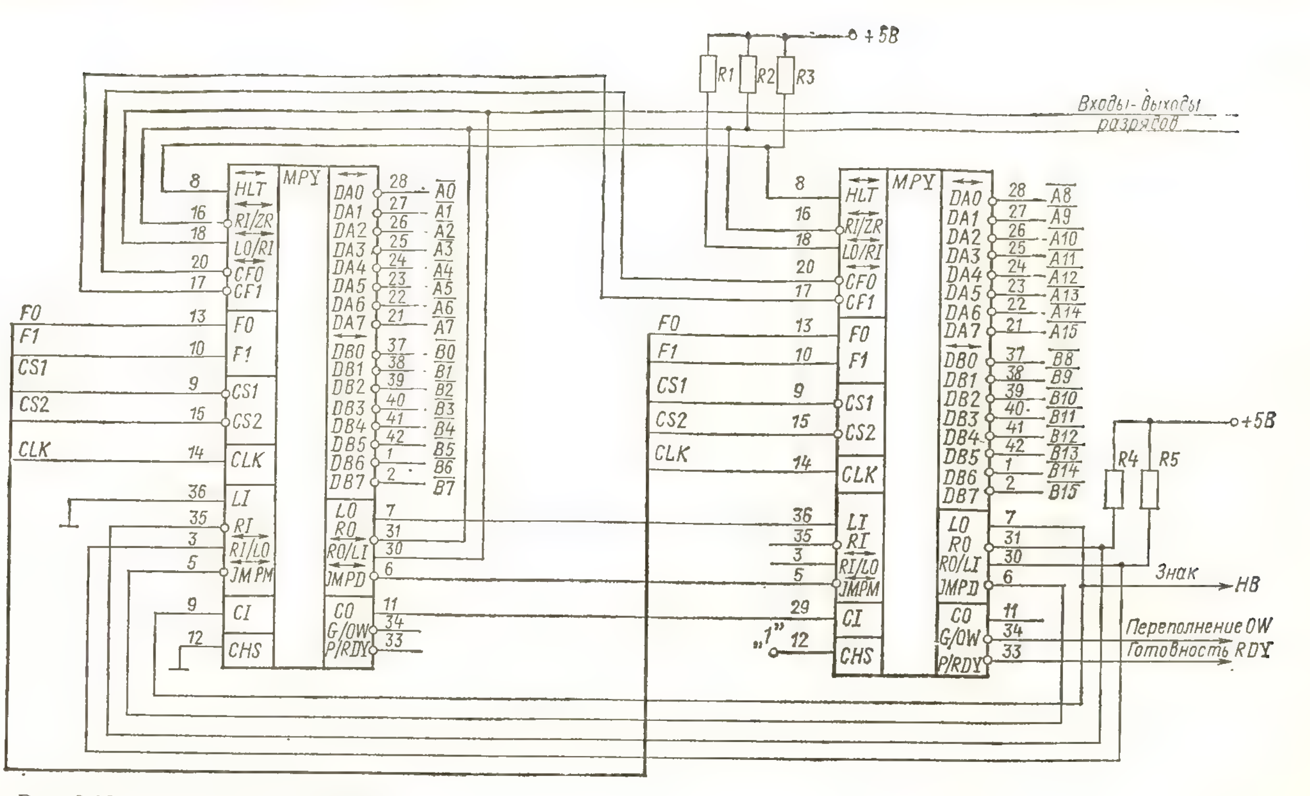


Рис. 8.16. Схема наращивания УМН

Таблица 8.17. **Соединение нескольких микросхем КР1802ВР2**

Номер вывода	Обозначение	Правила соединения
8	*HLT*	С выводами *HLT* остальных БИС
16	*RI/ZR*	С выводами *RI/ZR* остальных БИС и выводом *R*0 самой младшей БИС
18	*LO/RI*	С выводами *LO/RI* остальных БИС и выводом *RO/LI* самой младшей БИС
30	*RO/LI*	С выводом *RI/LO* более младшей БИС. У самой младшей БИС — с выводами *LO/RI* остальных БИС
31	*R*0	С выводом *RI* более младшей БИС. У самой младшей БИС — с выводами *RI/ZR* остальных БИС
7	*L*0	С выводом *LI* более старшей БИС. У самой старшей — с выводом *CI* самой младшей БИС
35	*RI*	С выводом *R*0 более старшей БИС. У самой старшей БИС не используется
36	*LI*	С выводом *L*0 более младшей БИС. У самой младшей БИС заземляется
3	*RI/L*0	С выводом *R*0/*LI* более старшей БИС. У самой старшей БИС не используется
5	*JMPM*	С выводом *JMPD* более старшей БИС. У самой старшей с выводом *JMPD* самой младшей БИС
6	*JMPD*	С выводом *JMPM* более младшей БИС. У самой младшей с выводом *JMPM* самой старшей БИС
20	*CF*0	С выводом *CF*0 остальных БИС
17	*CF*1	С выводом *CF*1 остальных БИС
29	*CI*	Подается значение выходного переноса из более младшей БИС или со схемы ускоренного переноса. У самой младшей БИС — с выводом *L*0 самой старшей БИС
11	*C*0	Без использования схемы ускоренного переноса соединяется с выводом *CI* более старшей БИС
13	*F*0	С соответствующими выводами остальных БИС
10	*F*1	С соответствующими выводами остальных БИС
9	*CS*1	С соответствующими выводами остальных БИС
15	*CS*2	С соответствующими выводами остальных БИС
14	*CLK*	С соответствующими выводами остальных БИС

анализирована и выдвинута, *ТгАкт* обнуляется и выдается сигнал *JMPM*, запускающий *ТгАкт* в следующей БИС. В старшей БИС УМН при обнулении *ТгАкт* сигнал *JMPM* не выдается, а *УУ* вырабатывает сигнал *HLT*, который сообщает об окончании циклического участка алгоритма.

После окончания циклической части алгоритма в триггер знака множителя *ТгТ1* (см. рис. 8.14) записывается состояние старшего разряда множителя (равного в этот момент *У1*), представляющего собой знак множителя.

После этого, в связи с тем, что умножение проводилось в дополнительном коде, необходимо провести один шаг коррекции результата. Для коррекции на вход *B* сумматора подается содержимое *Р3*, сдвинутое на два разряда вправо. Этим устанавливается соответствие младших разрядов суммы частичных произведений младшим разрядам расширенный разрядной сетки.

Информация для входа *A* сумматора формируется в зависимости от кода операции и значений *ТгТ* и *ТгТ1* следующим образом.

В случае умножения кодов *ТгТ1* = 0 (так как умножаются положительные величины). В этом случае необходима коррекция в виде прибавления множимого, т. е. если имеется перенос из старшей пары разрядов множителя, на вход *A* сумматора подается прямой код Р2.

В случае умножения чисел, если множитель положителен и нет переноса из старшей пары, то на вход *A* сумматора подается 0. Если множитель отрицателен, то возможны два варианта:

1) отсутствие переноса; необходимо вычесть множимое, т. е. на вход *A* сумматора подается Р2;

2) перенос; следует прибавить множимое и его же вычесть.

После проведения коррекции результат записывается в Р3 и БИС УМН переходит в состояние «Готова», т. е. возможно выполнение любой операции из табл. 8.16.

Состояние «Готова» характеризуется следующими признаками результата:

переполнение; выдается на вывод *G/OW* старшей БИС УМН. Поскольку при умножении чисел переполнение не возникает, то $\overline{G}/OW$ = «0»;

расширение, равное нулю; если какой-либо разряд Р3 отличен от старшего разряда Р1 в старшей БИС и единице в противоположном случае, подается на вывод *LO/RI* и показывает, что произведение записано только в регистре Р1, а в Р3 записаны только не значащие цифры;

нуль результата, равный единице; если Р3 = Р1 = 0, т. е. результат, записанный в данной БИС, равен нулю, выдается на вывод *RI/ZR*;

знак результата, равный значению старшего разряда Р3 при умножении чисел и нулю при умножении кодов, выдается на вывод *L*0. Значение 7-го разряда Р1 выдается на вывод *CF1*.

Операция «Деление кодов» инициируется так же как и умножение сигналом $\overline{CS1}$. БИС УМН переходит в состояние «Не готова», фиксирует код операции и разрешает прием младшего слова делимого и делителя в регистры Р1 и Р2 с магистралей *DA* и *DB* соответственно. Делению должна предшествовать операция «Загрузка».

В УМН реализовано деление по циклическому алгоритму без восстановления остатка, причем старшее слово делимого должно быть меньше делителя, иначе возникает переполнение.

При делении первый бит частного записывается в ТгТ и выдается на вывод *G/OW* старшей БИС УМН.

В каждом цикле деления один бит частного определяется по знаку очередного остатка. Знак остатка выдается на вывод *C*0 старшей БИС.

В первом цикле на вход A сумматора подается обратный код Р2, на вход B — прямой код Р3. Единица для получения дополнительного кода из обратного подается как и при умножении с вывода $L0$ старшей БИС.

Значение $S8$ записывается в ТгТ и определяет переполнение, если $S8 = 0$. Остаток, полученный при вычитании, записывается в Р3.

В каждом последующем цикле, если остаток отрицателен и необходимо провести сложение ($\overline{CF0} = 0$), подается прямой код Р2, если остаток положителен ($\overline{CF0} = 1$) — обратный код Р2.

На вход сумматора подается сдвинутое влево на один разряд содержимое Р3. Старший разряд Р3 при делении постоянно подается на вывод RI/LO. В младший разряд входа B сумматора подается бит, поступающий с вывода RO/LI. У младшей БИС на этот вывод поступает при делении бит, выдвинутый из старшего разряда Р1 той БИС, у которой ТгАкт = 1. Как и при умножении, Р1 сдвигается влево только у той БИС, у которой ТгАкт = 1. Первоначально ТгАкт устанавливается в 1 в старшей БИС по сигналу $\overline{CS1}$. Через восемь циклов, в течение которых часть делимого, записанная в Р1 данной БИС, полностью выдвигается и заменяется разрядами частного, ТгАкт обнуляется и выдается сигнал $\overline{JMPD}$, который, будучи подан на вывод $JMPM$ следующей (более младшей) БИС, запускает ее ТгАкт. По сигналу $\overline{JMPM}$, который поступает в старшую БИС УМН с вывода $JMPD$ младшей БИС, выдается сигнал HLT, который извещает об окончании циклического участка алгоритма.

Далее следует заключительный шаг алгоритма — восстановление остатка. При этом на вход A сумматора в зависимости от знака последнего остатка подается Р2 (отрицательный остаток) или код нуля (положительный остаток). На вход B сумматора подается Р3. Полученный остаток записывается в Р3. После завершения восстановления остатка БИС УМН переходит в состояние «Готова».

В состоянии «Готова» выдаются следующие признаки результата:

переполнения, равный значению ТгТ; выдается на вывод старшей БИС УМН;

расширения, равный единице, выдается на вывод LO/RI;

нуля результата, равный единице; если Р1 = 0, выдается на вывод RI/ZR;

знак результата, равный нулю, выдается на вывод $L0$.

Все операции «Чтение» инициируются сигналом $\overline{CS2}$. Чтение результатов, хранящихся в регистрах Р1 и Р3, на магистрали DA и DB определяется кодом операции на выводах $F0$ и $F1$. Результаты хранятся на регистрах Р1 и Р3 до записи в УМН новой информации по сигналу $\overline{CS1}$, т. е. допускается их много-

кратное чтение. После окончания операций чтения выдаются признаки результата, выработанные в операции, предшествующей чтению.

Статические параметры приведены в табл. 8.18. Временна́я диаграмма работы БИС УМН дана на рис. 8.17.

Диаграмма приведена при работе одной схемы (умножение 8-разрядных операндов), выводы схемы соединены согласно рис. 8.15 и табл. 8.17. На шинах DA, DB будет результат умножения при соблюдении временны́х соотношений, показанных

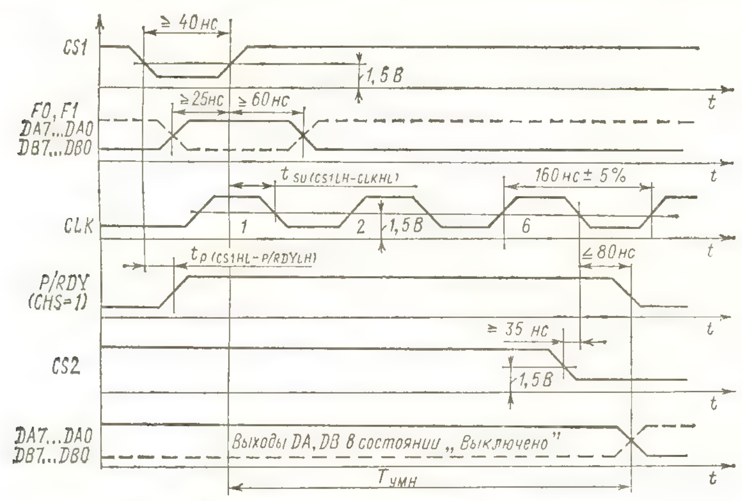


Рис. 8.17. Временна́я диаграмма работы УМН

на рис. 8.17, и обеспечении выдачи информации на шинах DA, DB ($F0 = F1 = \overline{CS2} =$ «0»). Минимальное гарантированное время умножения 940 нс.

В общем случае время умножения ($T_{\text{умн}}$) и время деления ($T_{\text{дел}}$) определяются по следующим формулам:

$$T_{\text{умн}} = (4n + 1)T + \tau_{WPH} + t_{p\,(CLKL-P/RDY)} + t_\Delta;$$

$$T_{\text{дел}} = (8n + 1)T + t_{p\,(CLKL-P/RDY)} + t_\Delta,$$

где T — период следования синхроимпульсов; n — число объединенных БИС УМН; τ_{WPH} — длительность положительного синхроимпульса; $t_{p\,(CLKL-P/RDY)}$ — время задержки распространения от синхроимпульса до признака «Готов»; t_Δ — дополнительное время, учитывающее возможную несинхронность сигналов $\overline{CS1}$ и CLK ($\tau_\Delta = 0 \ldots T$).

Период следования синхроимпульсов при увеличении разрядности операндов возрастает следующим образом:

$T \geqslant 160$ нс для одной БИС УМН;

$T \geqslant 220$ нс для двух БИС УМН;

Таблица 8.18. Статические параметры микросхемы КР1802ВР2

Параметр, единица измерения	Обозначение	Значения		Режим измерения
		мин.	макс.	
Входной ток «0», мА, для входов:	I_{IL}			U_{CC} = 5,25 В U_{IL} = 0,45 В U_{IH} = 2,4 В
RI, RI/L0, CI, CLK, CHS		—0,25	—	
DB7 ... DB0		—0,4	—	
DA7 ... DA0, F0, F1, L0/RI, RI/ZR		—0,45	—	
LI, RO/LI, CS2		—0,5	—	
JMPD		—0,8	—	
JMPM		—1,3	—	
HLT		—1,5	—	
CF0, CF1, CS1		—2,0	—	
Входной ток «1», мкА, для входов:	I_{IH}			U_{CC} = 5,25 В U_{IH} = 5,25 В U_{IL} = 0,5 В
CI, CLK, CHS, RI, CS1, RI/L0		—	40	
F0, F1, CS2, LI, L0/RI, RO/LI		—	80	
HLT		—	140	
Выходное напряжение «0», В, для входов:	U_{OL}			U_{CC} = 4,75 В, U_{IH} = 2,4 В U_{TH} = 2,0 В, U_{TL} = 0,8 В
G/OW, P/RDY, DA7 ... DA0				
DB7 ... DB0, CF0, CF1		—	0,5	I_{OL} = 15 мА
HLT		—	0,5	I_{OL} = 10,5 мА
RO/LI, L0/RI		—	0,5	I_{OL} = 4 мА
RI/ZR, R0		—	0,5	I_{OL} = 3,5 мА
JMPD		—	0,5	I_{OL} = 1,5 мА
JMPM		—	0,5	I_{OL} = 1,0 мА
C0, L0, RI/L0		—	0,5	I_{OL} = 0,5 мА
Выходное напряжение «1», В, для всех выходов (кроме выходов с открытым коллектором)	U_{OH}	2,4	—	U_{CC} = 4,75 В I_{OH} = —1,0 мА
Ток выключенного состояния, мкА, для выходов с тремя состояниями и открытым коллектором по выходам:	I_{OZ}			U_{CC} = 5,25 В
HLT		—	200	
остальным		—	100	
Ток короткого замыкания на выходе (кроме выходов с открытым коллектором), мА	I_{OS}	—50	—5	U_{CC} = 5,25 В U_{OH} = 0 В
Ток потребления, мА	I_{CC}	200	300	U_{CC} = 5,25 В

$T \geqslant (160 + (n - 1)\ 60)$ нс при объединении более двух БИС УМН без схемы ускоренного переноса (СУП), где $n = 1, 2, 3 \ldots$;

$T \geqslant (220 + T_{СУП})$ нс при объединении более двух БИС со схемой ускоренного переноса, где $T_{СУП}$ — время задержки распространения от входов Pi, Gi СУП до выходов переноса СУП определяется конкретным типом СУП (например, К589ИК03, К1804ВР1).

8.6. БИС ИНТЕРФЕЙСА КР1802ВВ2

БИС И предназначена для управления обменом информацией между устройствами в одно- или многопроцессорной вычислительной системе с асинхронной дисциплиной обмена.

БИС И выполняет 11 инструкций в соответствии с кодом, поступающим на управляющие входы *F*2 ... *F*0 и *H*. БИС И выполняет арбитраж запросов на обмен информацией после окончания предыдущего акта обмена.

Тактовая частота работы БИС И — до 10МГц. Условное графическое обозначение БИС И приведено на рис. 8.18. В табл. 8.19 дано назначение выводов. Структурная схема БИС И представлена на рис. 8.19.

Рис. 8.18. Условное графическое обозначение БИС И

Основу структурной схемы БИС И составляют:

регистр инструкций *РИ*, буферизирующий информацию по линиям *F*2 ... *F*0 и *H*;

дешифратор инструкций *ДШ*, формирующий сигналы в соответствии с кодом операции;

устройство управления *УУ* выдает и принимает сигналы управления обменом;

регистр сдвига *PC*, формирующий по каждому фронту импульса синхронизации импульс на шину *Q* в *УУ*;

триггера готовности, запроса цикла, обмена и цикла *ТГ*, *ТЗЦ*, *ТО*, *ТЦ* соответственно, обеспечивающие бессбойное функционирование основных узлов БИС И;

линия задержки *ЛЗ*, исключающая возможность появления «гоночных» ситуаций при реализации алгоритма арбитража;

регистр хранения запросов «главного» процессора *РХЗГ*.

Функционирование БИС И начинается с подачи на входы *F*0, *F*1, *F*2, *H* кода инструкции (табл. 8.20). Эта информация поступает на входы *РИ* и защелкивается на этом регистре при низком уровне сигнала на входе *WF*. Затем происходит дешифрация кода

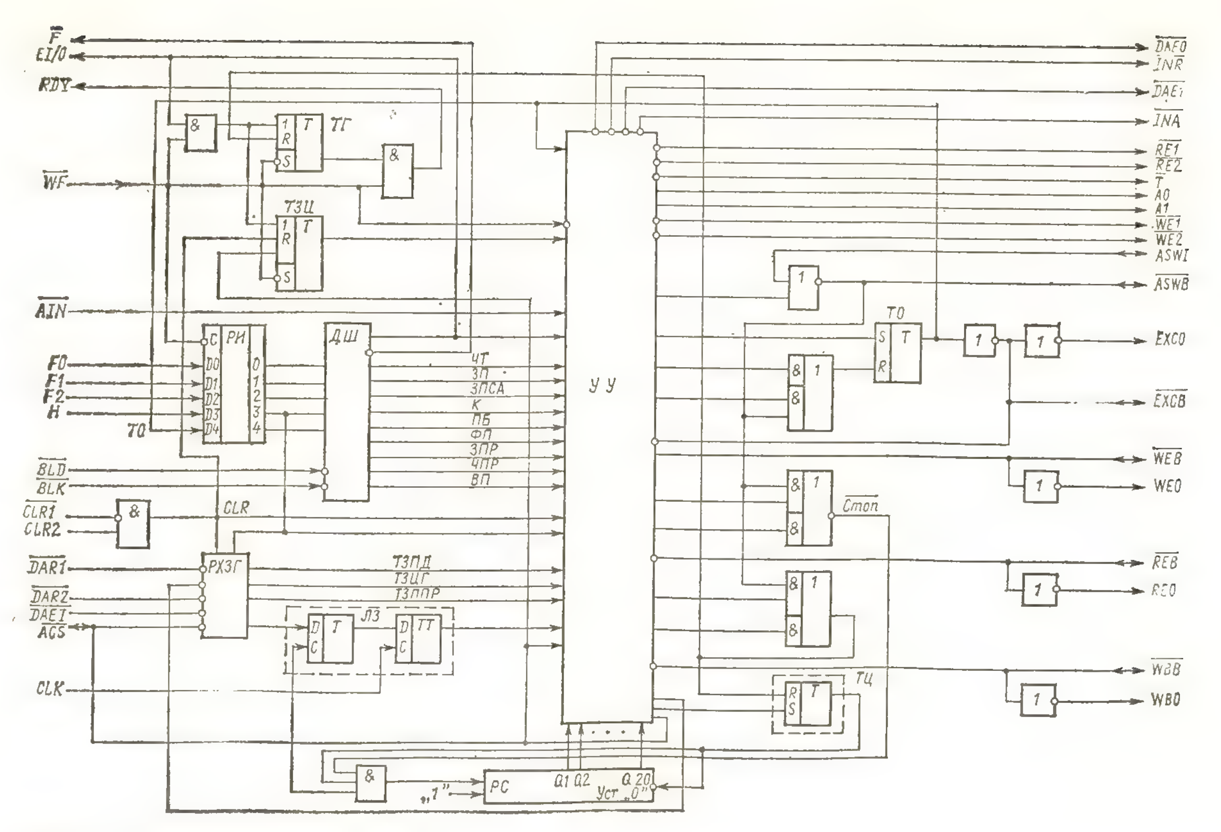


Рис. 8.19. Структурная схема БИС И

Таблица 8.19. **Назначение выводов микросхемы КР1802ВВ2**

Номер вывода	Назначение	Разряд	Обозначение	Тип
1	Сигнал «Обмен»	—	*EXCB*	Двунаправленный
2	Приемник сигнала «Обмен»	—	*EXCO*	Вход
3	Адрес	0	*A*0	Выход
4	Запрос прерывания	—	*INR*	Выход
5	Разрешение прерывания — источник	—	*INA*	Выход
6	Сигнал «Ответ»	—	*ASWB*	Двунаправленный
7	Передатчик «Ответ»	—	*ASWI*	Вход
8	Приемник «Разрешение записи»	—	*WEO*	Выход
9	Разрешение записи	—	*WEB*	Двунаправленный
10	Приемник	—	*WBO*	Выход
11	Общий	—	*GND*	—
12	Запись байта	—	*WBB*	Двунаправленный
13	Импульс «Чтение 2»	—	*RE*2	Выход
14	Разрешение чтения	—	*REB*	Двунаправленный
15	Приемник «Разрешение чтения»	—	*REO*	Выход
16	Импульс «Чтение 1»	—	*RE*1	Выход
17	Импульс «Запись 1»	—	*WE*1	Выход
18	Сброс 1	—	*CLR*1	Вход
19	Сброс 2	—	*CLR*2	Вход
20	Адрес	1	*A*1	Выход
21	Импульс «Запись 2»	—	*WE*2	Выход
22	Синхронизация	—	*T*	Выход
23	Синхронизация	—	*CLK*	Вход
24	Разрешение прерывания — приемник	—	*AIN*	Вход
25	Фиксация пути	—	*F*	Выход
26	Блокировка ЧТ/ЗП данных	—	*BLD*	Вход
27	Разрешение внутреннего обмена	—	*EI/O*	Выход
28	Запись микроинструкции	—	*WF*	Вход
29	Блокировка ЧТ команды	—	*BLK*	Вход
30	Готовность	—	*RDY*	Выход
31	Главный процессор	—	*H*	Выход
32	Напряжение питания	—	U_{CC}	—
33	Шина микроинструкции	2	*F*2	Вход
34	Шина микроинструкции	1	*F*1	Вход
35	Шина микроинструкции	0	*F*0	Вход
36	Запрос прямого доступа	—	*DAR*1	Вход
37	Запрос прямого доступа от периферийного процессора	—	*DAR*2	Двунаправленный
38	Общий	—	*GND*	—
39	Подтверждение выборки	—	*ACS*	Двунаправленный

Номер вывода	Назначение	Разряд	Обозначение	Тип
40	Разрешение прямого доступа	—	$DAE1$	Выход
41	«Разрешение прямого доступа периферийному процессору» — приемник	—	$DAEI$	Вход
42	«Разрешение прямого доступа — периферийному процессору» — источник	—	$DAEO$	Выход
Примечание. Выходы $EXCB$, REB, WEB, $ASWB$, WBB, $DAR2$, ACS, RDY, T, $WE1$, $WE2$, $RE1$, $RE2$, $DAE1$, INR, $DAEO$, INA с открытым коллектором.				

инструкции и выдача сигналов управления на $УУ$ и на внешние выводы (F, EI/O).

После этого БИС И реализует один из девяти жестко заданных алгоритмов управления обменом в соответствии с табл. 8.20. Для каждого алгоритма регистр РС выдает на линии $Q1$ ... $Q20$ последовательность управляющих сигналов «Q», необходимых для нормального функционирования БИС. Временна́я диаграмма работы РС приведена на рис. 8.20.

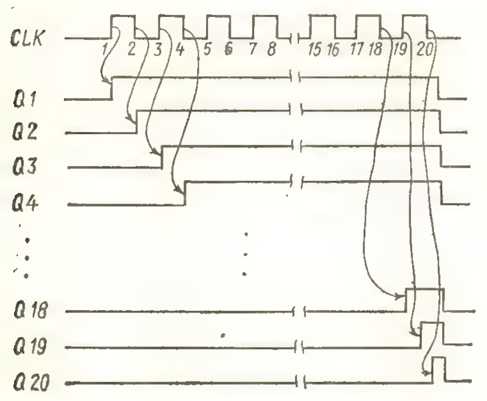


Рис. 8.20. Временна́я диаграмма работы РС

Регистр РХЗГ состоит из триггеров ТЗПД, ТЗЦГ и ТЗППР, работа которых описана в алгоритме арбитража.

При выполнении некоторых инструкций БИС И переходит в состояние ожидания сигнала ответа. «Ожидание» осуществляется посредством специальной схемы, которая вырабатывает сигнал СТОП.

РС прекращает выполнять сдвиг, если $\text{СТОП} = \text{ЧТ}\cdot\overline{ASWB}\times$ $\times\, Q8\cdot Q13 + \text{ЗП}\cdot\overline{ASWB}\cdot Q12\cdot\overline{Q15} + \text{ЗПСА}\cdot\overline{ASWB}\cdot Q5\cdot\overline{Q8} +$ $+\, \text{ЧПР}\cdot ASWB\cdot Q6\cdot\overline{Q8} + \text{ЗПР}\cdot REB\cdot Q3 + \text{ВП}\cdot REB\cdot Q1 = 1$.

При СТОП = 0 регистру сдвига разрешено функционирование.

Перед выполнением очередной инструкции БИС И осуществляет анализ запросов на обмен для схем с признаком «Главная»,

Таблица 8.20. **Инструкции, выполняемые БИС И**

Инструкция	Мнемоника	Входные сигналы ДШ					Выходные сигналы ДШ									
		$F0$	$F1$	$F2$	H	TO	ЧТ	ЗП	ЗПСА	К	ПБ	ФП	ЗПР	ЧПР	АПР	ВП
Чтение команды	ЧТК	0	0	0	X	0	1	0	0	1	0	0	0	0	0	0
Чтение слова	ЧТС	0	0	1	X	0	1	0	0	0	0	0	0	0	0	0
Чтение слова с фиксацией пути	ЧТФП	0	1	0	X	0	1	0	0	0	0	1	0	0	0	0
Запись слова	ЗПС	0	1	1	X	0	0	1	0	0	0	0	0	0	0	0
Запись слова после ЧТФП	ЗСФП	0	1	1	X	1	0	0	1	0	0	0	0	0	0	0
Запись байта	ЗПБ	1	0	0	X	0	0	1	0	0	1	0	0	0	0	0
Запись байта после ЧТФП	ЗБФП	1	0	0	X	1	0	0	1	0	1	0	0	0	0	0
Инициирование прерывания с передачей вектора	ЗПР	1	0	1	0	0	0	0	0	0	0	0	1	0	0	0
Чтение прерывания (прием вектора)	ЧПР	1	0	1	1	0	0	0	0	0	0	0	0	1	0	0
Адресное прерывание	АПР	1	1	0	X	0	0	0	0	0	0	0	0	0	1	0
Выдача пассивная	ВП	1	1	1	X	0	0	0	0	0	0	0	0	0	0	1

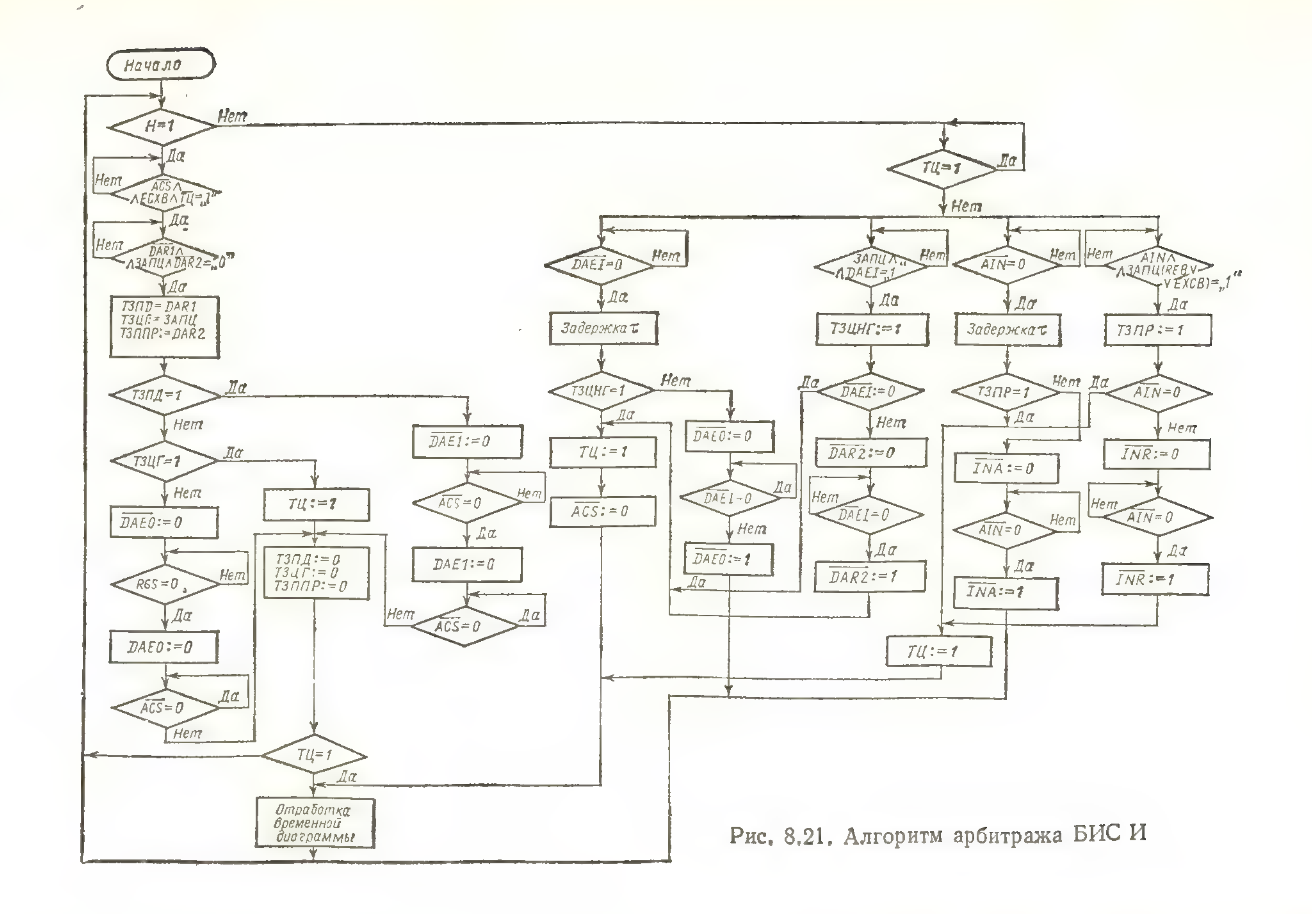


Рис. 8.21. Алгоритм арбитража БИС И

Таблица 8.21. **Статические параметры микросхемы КР1802ВВ2**

Параметр, единица измерения	Обозначение	Значения параметров		Режим измерения
		мин	макс.	
Входной ток «0», мА	I_{IL}	—0,25	—	U_{CC} = 5,25 В U_{IL} = 0,45 В
Входной ток «1», мкА	I_{IH}	—	40	U_{CC} = 5,25 В U_{IH} = 5,25 В
Выходной ток «1», мкА	I_{OH}	—	100	U_{CC} = 5,25 В
Выходное напряжение «0», В, по выходам:	U_{OL}			U_{CC} = 4,75 В
EXCO, *A*0, *WEO*, *WBO*, *RE*0, *RE*1, *RE*2, *WE*1, *WE*2 *A*1, *T*, *F*, *EI/O*, *RDY*		—	0,5	I_{OL} = 15 мА
EXCB, *INR*, *INA*, *WEB*, *WBB*, *REB*, *DAR*2, *ACS*, *DAEO*, *DAE*1, *ASWB*		—	0,6	I_{OL} = 60 мА
Выходное напряжение «1», В	U_{OH}	2,4	—	U_{CC} = 4,75 В I_{OH} = —1 мА
Ток потребления, мА	I_{CC}	—	250	U_{CC} = 5,25 В

а также запросов или сигналов разрешения для схем с признаком «Не главная». Алгоритм арбитража поясняется рис. 8.21.

Алгоритм включает две основные ветви: «Главная» и «Не главная». Выход на соответствующую ветвь осуществляется после анализа сигнала на входе *H* БИС И.

При наличии на входе *H* «1» БИС анализирует выполнение условия $ACS \wedge EXCB \wedge$ ТЦ = «1». Если условие не выполнено, то БИС переходит в режим ожидания. При выполнении этого условия БИС ожидает активных уровней сигналов («0») на одном из входов *DAR*1 и *DAR*2, либо выполнения условия $WF \wedge EI/O \wedge$ $\wedge$ ТЗЦ $\wedge$ ЗПR. Для компактности записи этого условия обозначим его «ЗАПЦ». Далее на триггерах ТЗПД, ТЗЦГ и ТЗППР запоминаются значения *DAR*1, ЗАПЦ и *DAR*2 соответственно. Затем осуществляется опрос этих триггеров на наличие. При наличии «1» в ТЗПД БИС формирует сигнал разрешения $\overline{DAE1}$ = «0» а затем, при *ACS* = «0», сбрасывается сигнал разрешения ($\overline{DAE1}$ = «1») и после сброса сигнала *ACS* = «1» выполняется об-

щая часть алгоритма. При этом сбрасываются в нуль триггера ТЗПД, ТЗЦГ и ТЗППР. Затем, при наличии «1» в триггере ТЦ, БИС переходит к выполнению очередной инструкции.

При наличии «1» на ТЗЦГ устанавливаются в «1» ТЦ и затем выполняется общая часть алгоритма, описанная выше. При установке в «1» ТЗППР устанавливается сигнал $DAEO$ = «0», при $\overline{ACS}$ = «0» сбрасывается сигнал $DAEO$ = «1». После сброса сигнала $\overline{ACS}$ = «1» осуществляется переход на общую часть алгоритма. При наличии «0» на входе H в БИС реализуется вторая ветвь алгоритма. Условием для ее выполнения является «0» в триггере ТЦ. Эта ветвь содержит четыре части, выполнение каждой из них определяется соответствующим условием: 1. $DAE1$ = «0»; 2. ЗАПЦ $\wedge$ $DAE1$ = «1»; 3. AIN = «0»; 4. AIN $\wedge$ ЗАПП $\wedge$ $(REB \vee EXCB)$ = 1.

Выполнение условий 1 и 3 исключает выполнение условий 2 и 4. Во избежание «гоночных» ситуаций в структуру БИС введены линии задержки. На рис. 8.21 достаточно подробно показана реализация алгоритма арбитража. Рассмотрим только одну (левую) часть ветви «Не главная». При удовлетворении условия $DAE1$ = = «0» через задержку τ опрашивается наличие «1» в ТЗЦНГ. После этого в «1» устанавливаются ТЦ и сигнал $\overline{ACS}$ = 0, затем отрабатывается очередная инструкция. При невыполнении условия ТЗЦНГ = «1» устанавливается сигнал $DAEO$ = O и после сброса сигнала $DAE1$ = «1» сбрасывается сигнал $DAEO$ = «1».

Статические параметры БИС И приведены в табл. 8.21.

8.7. СХЕМА ОБМЕНА ИНФОРМАЦИЕЙ КР1802ВВ1

БИС ОИ предназначена для использования в системах обработки цифровой информации в качестве регистровой памяти, имеющей в своем составе четыре 4-разрядных, раздельно адресуемых регистра. БИС позволяет вести обмен информацией по четырем шинам данных и любым из 4-х регистров. Кроме этого, внутренняя структура схемы предоставляет возможность организации на одном регистре счетчика внешних импульсов. БИС ОИ вырабатывает сигналы переноса в следующую тетраду (при работе регистра в счетном режиме), сигнал сравнения содержимого двух внутренних регистров. БИС ОИ позволяет производить «перекрестные» пересылки с одной шины на другую.

Условное графическое обозначение БИС ОИ представлено на рис. 8.22, назначение выводов — в табл. 8.22.

Структурная схема БИС ОИ (рис. 8.23) состоит из следующих устройств: четырех регистров $RG0 \ldots RG3$; дешифраторов $ДША$, $ДШВ$, $ДШС$, $ДШХ$, с помощью которых выбирается один из четырех регистров для обмена с соответствующей шиной; усилителей — формирователей сигналов разрешения обмена $УСФЕ$; усилителей записи $УСЗ$; мультиплексоров шин A, B, C, X — MA, MB, MC, MX; схемы сравнения содержимого $RG0$ и $RG3$ — $CxCP$.

БИС ОИ может вести обмен информацией по четырем магистралям *A*, *B*, *C*, *X*. Каждый из четырех внутренних регистров может функционировать в двух основных режимах: «Запись» и «Чтение». Режим записи информации в регистр *RG*0 ... *RG*3 с магистрали *A* реализуется подачей на входы *AA*0 и *AA*1 двоичного кода номера регистра на входы *ECA* и *WA* — сигналов «0».

При этом в выбранный регистр перепишется информация с шины *A*. Режим чтения на шину *A* реализуется подачей на входы *AA*0, *AA*1 кода выбранного регистра, а на входы *ECA* и *RA* — сигналов «0». При этом содержимое выбранного регистра поступит на шину *A*.

Работа с шинами *B*, *C*, *X* происходит аналогичным образом

Следует отметить, что при выборе одного и того же регистра при разрешении обмена с несколькими шинами в режиме чтения содержимое этого регистра будет выдаваться на все выбранные шины.

С другой стороны, при выборе одного и того же регистра и разрешении обмена по нескольким шинам в режиме записи в этот регистр будет записываться результат операции поразрядного логического сложения информации с выбранных шин.

Перекрестные коммутации шин можно производить записью в регистр с шины источника с последующим чтением информации с этого регистра в шину приемника.

Необходимо остановиться на особенностях работы регистра *RG*0. В дополнение к перечисленным возможностям внутренних регистров БИС *RG*0 может работать в режиме счетчика (инкрементного) с параллельной загрузкой информации. Счетные импульсы подаются на вход *CI*, и при переходе из «0» в «1» содержимое *RG*0 увеличивается на единицу. Параллельная загрузка информации в *RG*0 осуществляется при перепаде на входе разрешения записи с одной из магистралей из «0» в «1» с фронтом не более 150 нс. Однако не допускается одновременная загрузка *RG*0 с подачей на вход *CI* счетных импульсов.

Рис. 8.22. Условное графическое обозначение БИС ОИ

БИС ОИ непрерывно вырабатывает признак результата сравнения содержимого регистра *RG*0 и *RG*3 с подачей его на вывод *F*.

При наличии кода 1111 в регистре *RG*0 и поступлении положительного импульса на вход *CI* БИС вырабатывает сигнал выходного переноса *CO*. Статические и динамические параметры БИС ОИ приведены в табл. 8.23, 8.24 соответственно.

На рис. 8.24—8.28 даны временные диаграммы работы БИС ОИ.

Рис. 8.23. Структурная схема БИС ОИ

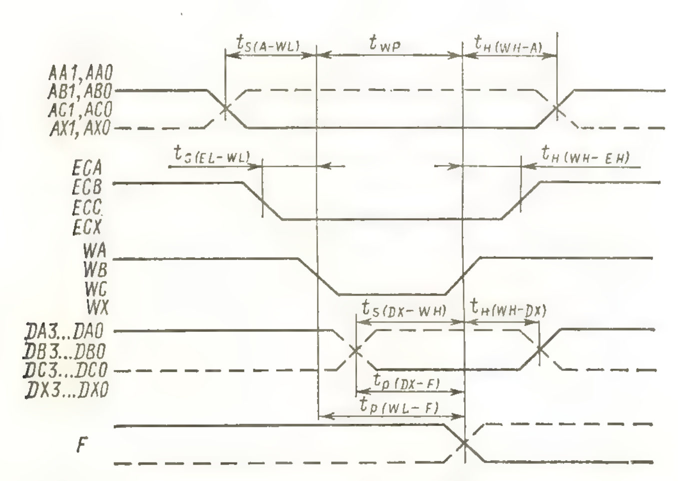


Рис. 8.24. Временна́я диаграмма БИС ОИ при записи с шин *A*, *B*, *C*, *X* в *RG1* — *RG3*

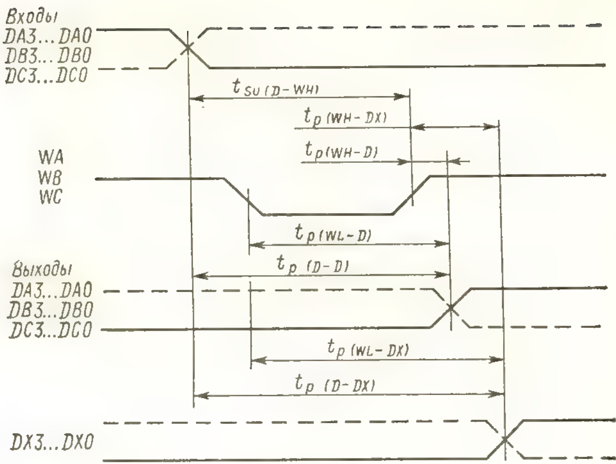


Рис. 8.25. Временная диаграмма БИС ОИ в режиме «Запись—чтение» при одинаковой информации на шинах *A*, *B*, *C*, *X* (запись не с шины *X*)

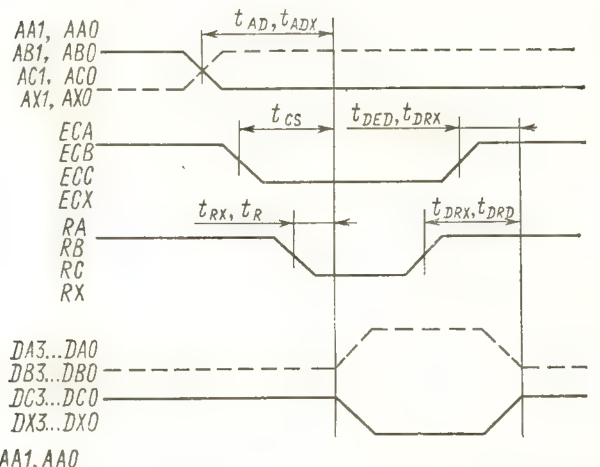


Рис. 8.26. Временная диаграмма БИС ОИ при чтении на шины *A*, *B*, *C*, *X*

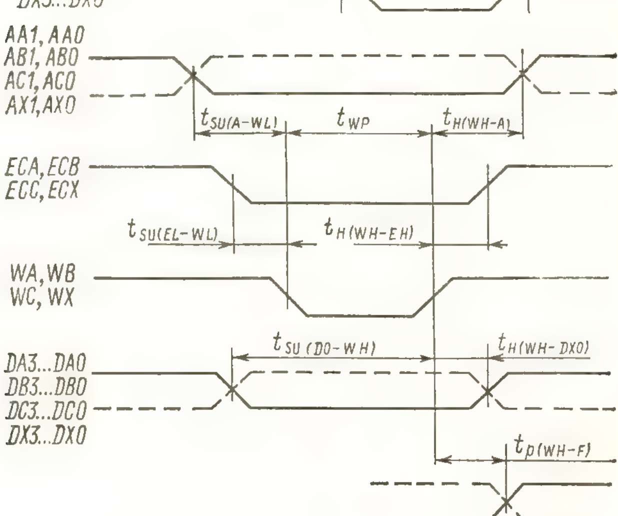


Рис. 8.27. Временная диаграмма БИС ОИ при записи информации с шин *A*, *B*, *C* в *RG*0

Таблица 8.22. **Назначение выводов микросхемы КР1802ВВ1**

Номер вывода	Назначение	Разряд	Обозначение	Тип
1	Адрес регистра канала *B*	0	*AB*0	
2	Адрес регистра канала *C*	1	*AC*1	
3		0	*AC*0	Вход
4	Адрес регистра канала *X*	1	*AX*1	
5		0	*AX*0	
6	Шина данных канала *C*	3	*DC*3	
7		2	*DC*2	
8	Шина данных канала *X*	3	*DX*3	Двунаправленный
9		2	*DX*2	
10		1	*DX*1	
11	Общий	—	*GND*	—
12	Шина данных канала *X*	0	*DX*0	
13	Шина данных канала *C*	1	*DC*1	Двунаправленный
14		0	*DC*0	
15	Считывание канала *C*	—	*RC*	
16	Считывание канала *X*	—	*RX*	
17	Разрешение обмена с каналом *X*	—	*ECX*	
18	Разрешение записи для канала *X*	—	*WX*	
19	Разрешение обмена с каналом *C*	—	*ECC*	
20	Разрешение записи для канала *C*	—	*WC*	
21	Разрешение записи для канала *B*	—	*WB*	Вход
22	Разрешение обмена с каналом *B*	—	*ECB*	
23	Разрешение записи для канала *A*	—	*WA*	
24	Разрешение обмена с каналом *A*	—	*ECA*	
25	Считывание канала *A*	—	*RA*	
26	То же, *B*	—	*RB*	
27	Перенос	—	*CI*	
28	Признак равенства содержимого *RG*0 и *RG*3	—	*F*	Выход
29	Шина данных канала *B*	0	*DB*0	
30	То же *A*	0	*DA*0	Двунаправленный
31	» *B*	1	*DB*1	
32	Напряжение питания	—	U_{cc}	—
33	Шина данных канала *A*	1	*DA*1	
34	То же *B*	2	*DB*2	
35	» *A*	2	*DA*2	Двунаправленный
36	» *B*	3	*DB*3	
37	» *A*	3	*DA*3	
38	Общий	—	*GND*	—
39	Перенос	—	*CO*	Выход
40	Адрес регистра канала *A*	1	*AA*1	Вход
41		0	*AA*0	
42	Адрес регистра канала *B*	1	*AB*1	Вход

Примечание. Выходы *DA*3 ... *DA*0, *DB*3 ... *DB*0, *DC*3 ... *DC*0 трехстабильные, выходы *DX*3 ... *DX*0, *F* с открытым коллектором.

Таблица 8.23. **Статические параметры микросхемы КР1802ВВ1**

Параметр, единица измерения	Обозначение	Значения параметров		Режим измерения
		мин.	макс.	
Входной ток «0», мА, для входов:	I_{IL}			U_{CC} = 5,25 В U_{IL} = 0,45 В
*AA*1, *AA*0, *AB*1, *AB*0, *AC*1, *AC*0, *AX*1, *AX*0		—0,75		
*DA*3 ... *DA*0, *DB*3 ... *DB*0, *DC*3 ... *DC*0, *DX*3 ... *DX*0, *WA*, *WB*, *WC*, *WX*, *RA*, *RB*, *RC*, *RX*		—0,25		
ECA, *ECB*, *ECC*, *ECX*		—0,5	—	
CI		—2,0	—	
Входной ток «1», мкА, для входов:	I_{IH}			U_{CC} = 5,25 В U_{IH} = 5,25 В
*AA*1, *AA*0, *AB*1, *AB*0, *AC*1, *AC*0, *AX*1, *AX*0		—	120	
*DA*3 ... *DA*0, *DB*3 ... *DB*0, *DC*3 ... *DC*0, *DX*3 ... *DX*0, *WA*, *WB*, *WC*, *WX*, *RA*, *RB*, *RC*, *RX*		—	40	
ECA, *ECB*, *ECC*, *ECX*		—	80	
CI		—	160	
Выходной ток «1», мкА, для выходов *DX*3 ... *DX*0, *F*	I_{OH}	—	100	U_{CC} = 5,25 В U_{TL} = 0,8 В U_{TH} = 2,0 В
Выходное напряжение «0», В, для выходов:	U_{OL}			U_{CC} = 4,75 В U_{TL} = 0,8 В U_{TH} = 2,0 В
*DA*3 ... *DA*0, *DB*3 ... *DB*0, *DC*3 ... *DC*0, *CO*, *F*		—	0,5	I_{OL} = 15 мА
*DX*3 ... *DX*0		—	0,8	I_{OL} = 60 мА U_{TL} = 1,4 В (для *DX*3 ... *DX*0)
Выходное напряжение «1», В, для выходов: *DA*3 ... *DA*0, *DB*3 ... *DB*0, *DC*3 ... *DC*0, *CO*	U_{OH}	2,4	—	U_{CC} = 4,75 В U_{TL} = 0,8 В U_{TH} = 2,0 В I_{OH} = —1 мА
Ток выключенного состояния, мкА	I_{OZ}	—	100	U_{CC} = 5,25 В U_{OH} = 5,25 В U_{TH} = 2 В U_{TL} = 0,8 В
Ток короткого замыкания, мА	I_{OS}	—65	—15	U_{CC} = 5,0 В U_{OL} = 0 В
Ток потребления, мА	I_{CC}	—	280	U_{CC} = 5,25 В

Таблица 8.24. **Динамические параметры микросхемы КР1802ВВ1**

Параметр	Обозначение	Значения, нс		
		мин.	ном.	макс.
Время выбора микросхемы	t_{CS}	—	25	37
Время считывания от адреса	t_{AD}	—	40	60
Время считывания от сигналов чтения	t_R	—	25	37
Время выключения выходов *DA*3 ... *DA*0, *DB*3 ... *DB*0, *DC*3 ... *DC*0 от входов *RA*, *RB*, *RC*	t_{DRD}	—	25	37
Время выключения микросхемы	t_{DED}	—	25	37
Время считывания от адреса *AX*1, *AX*0	t_{ADX}	—	60	80
Время считывания на выходы *DX*	t_{RX}	—	25	37
Время выключения от входа *RX*	t_{RDX}	—	25	37
Время установки сигнала относительно отрицательного фронта импульса записи на входах:				
AA, *AB*, *AC*, *AX*	$t_{SU\ (A-WL)}$	15	—	—
ECA, *ECB*, *ECC*, *ECX*	$t_{SU\ (EL-WL)}$	25	—	—
Длительность импульса записи	t_{WP}	50	30	—
Время удержания сигнала относительно положительного фронта импульса записи на входах:				
AA, *AB*, *AC*, *AX*	$t_{H\ (WH-A)}$	—	10	—
ECA, *ECB*, *ECC*, *ECX*	$t_{H\ (WH-EH)}$	—	15	—
Время установки информации на входах *DA*, *DB*, *DC* при записи в регистры *RG*1, *RG*2, *RG*3 относительно положительного фронта импульса записи	$t_{SU\ (D-WH)}$	—	10	—
Время удержания информации на входах *DA*, *DB*, *DC* при записи в регистры *RG*1, *RG*2, *RG*3 относительно положительного фронта импульса записи	$t_{H\ (WH-D)}$	—	15	—
Время задержки распространения сигнала от входов *DA*, *DB*, *DC* до выхода *F*	$t_{р\ (D-F)}$	—	45	80
То же от входа *DX* до выхода *F*	$t_{р\ (DX-F)}$	—	55	90
» от входа *CI* до выхода *CO*	$t_{р\ (CI-CO)}$	—	15	25
» от входов *DA*, *DB*, *DC* до выходов:				
DA, *DB*, *DC*	$t_{р\ (D-D)}$	—	40	60
DX	$t_{р\ (D-DX)}$	—	50	80
То же от входа *DX* до выходов *DA*, *DB*, *DC*	$t_{р\ (DX-D)}$	—	50	70
Время задержки распространения сигнала от отрицательного фронта импульса записи до выходов:				
F	$t_{р\ (WL-F)}$	—	55	80
DA, *DB*, *DC*	$t_{р\ (WL-D)}$	—	55	80
DX	$t_{р\ (WL-DX)}$	—	80	—
Время задержки распространения сигнала от положительного фронта импульса записи до выхода *F*	$t_{р\ (WH-F)}$	—	60	90

Параметр	Обозначение	Значения, нс мин.	ном.	макс.
Время задержки распространения сигнала от положительного фронта сигнала *CI* до выходов:				
DA, *DB*, *DC*	$t_{р\ (CIH-D)}$	—	50	80
DX	$t_{р\ (CIH-DX)}$	—	50	80
F	$t_{р\ (CIH-F)}$	—	50	80
Время задержки распространения сигнала от положительного фронта импульса записи (при записи в *RG*0) до выходов:				
DA, *DB*, *DC*	$t_{р\ (WH-D)}$	—	60	90
DX	$t_{р\ (WH-DX)}$	—	60	90
Время установки сигнала оносительно положительного фронта импульса записи на входах:				
DX	$t_{SU\ (DX-WH)}$	—	20	—
DA, *DB*, *DC* } при записи в *RG*0	$t_{SU\ (D0-WH)}$	—	5	—
DX } при записи в *RG*0	$t_{SU\ (DX0-WH)}$	—	5	—
То же относительно положительного фронта сигнала *CI* на входах:				
AA, *AB*, *AC*, *AX*	$t_{SU\ (A-CIH)}$	—	10	—
ECA, *ECB*, *ECC*, *ECX*	$t_{SU\ (EH-CIH)}$	—	30	—
WA, *WB*, *WC*, *WX*	$t_{SU\ (WH-CIH)}$	—	20	—
Время удержания сигнала относительно положительного фронта импульса записи на входах:				
DX	$t_{H\ (WH-DX)}$	—	5	—
DA, *DB*, *DC* } при записи в *RG*0	$t_{H\ (WH-D0)}$	—	20	—
DX } при записи в *RG*0	$t_{H\ (WH-DX0)}$	—	20	—
Время удержания сигнала относительно положительного фронта сигнала *CI* на входах:				
ECA, *ECB*, *ECC*, *ECX*	$t_{H\ (CIH-EL)}$	—	0	—
WA, *WB*, *WC*, *WX*	$t_{H\ (CIH-WL)}$	—	0	—
AA, *AB*, *AC*, *AX*	$t_{H\ (CIH-A)}$	—	0	—

Следует обратить внимание на то, что шины A, B, C могут использоваться для организации внутриплатных линий связи. Шина *X* имеет более высокую нагрузочную способность (до 60 мА) и может работать на длинную согласованную линию связи.

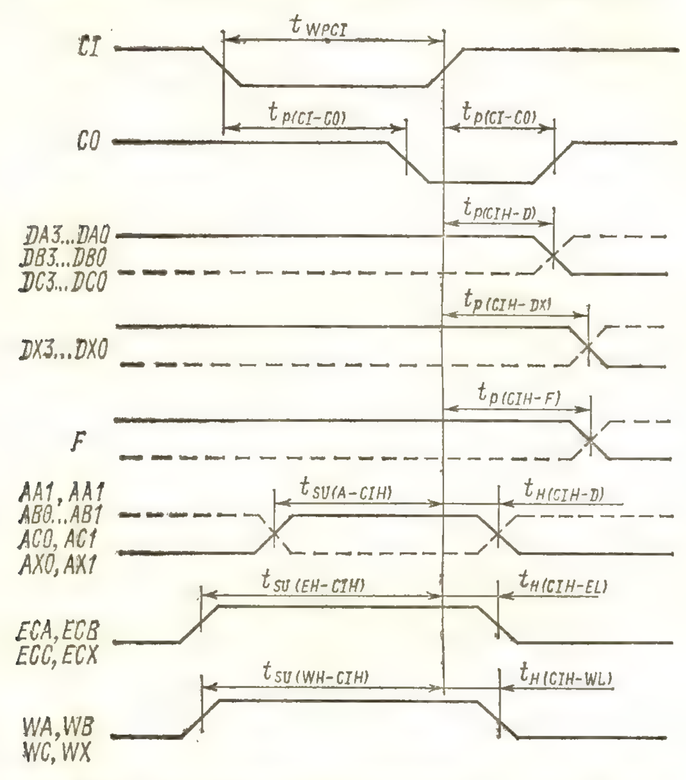


Рис. 8.28. Временная диаграмма БИС ОИ в режиме счета

8.8. ПРИМЕРЫ ПОСТРОЕНИЯ ПРОЦЕССОРОВ НА БАЗЕ БИС СЕРИИ КР1802

Ниже приведены структурные схемы двух процессоров и описания их работы. Один из них, процессор микроЭВМ «Электроника НЦ-05», выполняет команды семейства микроЭВМ НЦ. Другой выполняет команды подмножества системы команд ЕС ЭВМ.

Структурная схема процессора «Электроника НЦ-05» (рис. 8.29) состоит из следующих узлов: операционного блока *ОБ*; регистров общего назначения *РОН*; специальных регистров *СР*; блока обмена информацией *БОИ*; регистра состояния процессоров *РСП*; блока микропрограммного управления *БМУ*; блока приоритетного прерывания *БПП*.

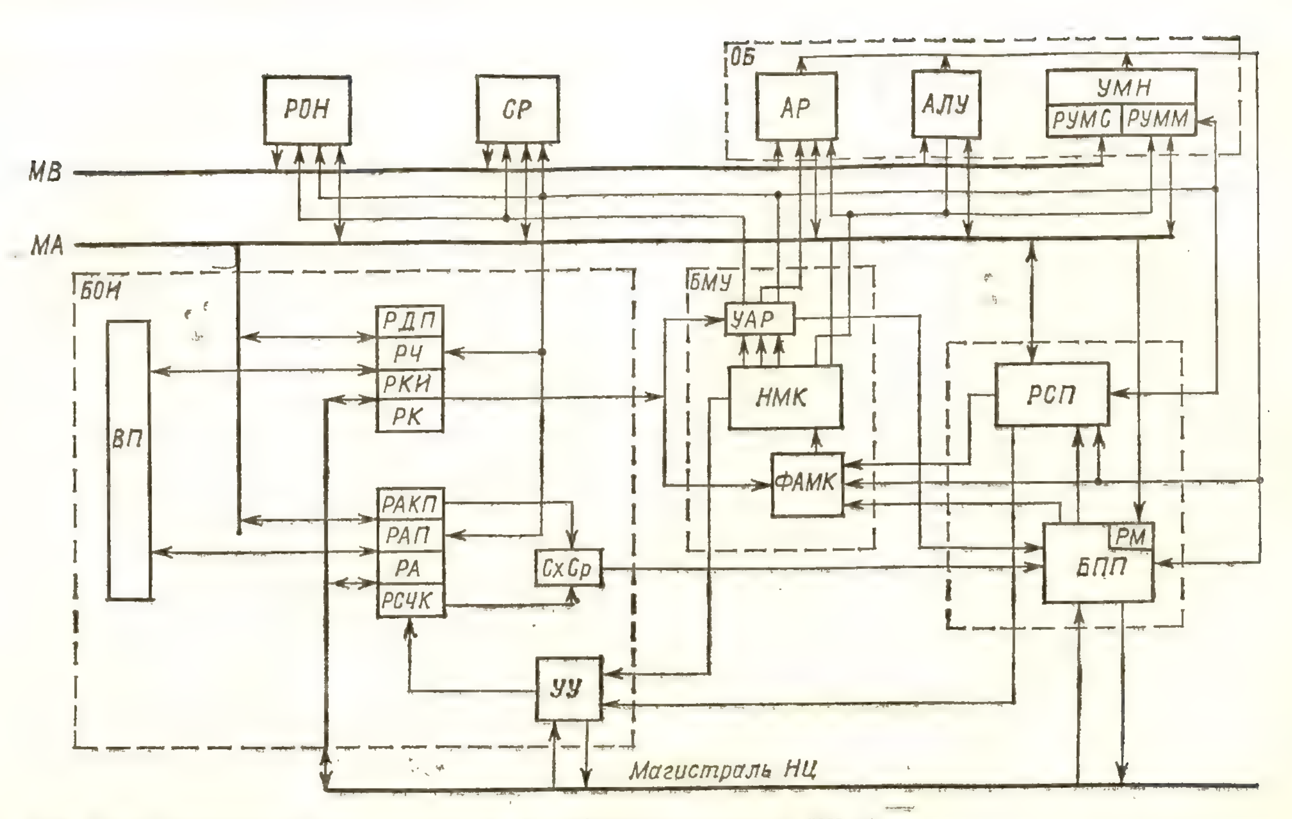


Рис. 8.29. Структурная схема процессора микроЭВМ «Электроника НЦ 05»

Операционный блок *ОБ* служит для организации пересылок между регистрами процессора и выполнения арифметико-логических операций над двумя операндами. *ОБ*, построенный на БИС КР1802ВС1, КР1802ВР1, КР1802ВР2, включает:

АЛУ, предназначенное для реализации операций с фиксированной и плавающей запятой;

арифметический расширитель *АР*, обеспечивающий выполнение операций сдвигов, нормализации;

умножитель *УМН*, выполняющий команды умножения и деления.

В *ОБ* команды поступают по магистралям *МА* и *МВ*, результат операции выдается по *МА*. Вместе с результатом выдаются признаки результата «Нуль», «Переполнение», «Расширение». Признак расширения используется для выполнения операций с двойной точностью.

В *АР* операнд поступает по магистрали *МА*, а параметр сдвига либо по *МВ*, либо из поля микрокоманды. При операциях расширенного сдвига используется внутренний регистр расширения *АР*, в котором запоминаются выдвинутые разряды.

При выполнении операций умножения произведение фиксируется на двух внутренних регистрах умножителя: регистре старшего *РУМС* и младшего *РУММ* слов умножителя. Оба регистра адресуются в поле микрокоманды, и их содержимое может быть считано по магистралям *МА* и *МВ*. В операциях деления старшее слово делимого загружается предварительно в *РУМС*. В *РУММ* фиксируется частное от деления, а в *РУМС* — остаток. Это позволяет производить последовательное деление многословного делимого на однословный делитель без дополнительных пересылок между регистрами, что позволяет существенно ускорить выполнение деления. Специфика работы УМН позволяет совмещать выполнение операций умножения и деления с работой узлов *АЛУ* и *АР*.

Блок *РОН*, построенный на БИС КР1802ИР1 содержит, 16 регистров, доступных программисту. Они предназначены для хранения промежуточных результатов операций, а также для формирования адресов при выполнении команд, имеющих различные типы адресации. Большое число внутренних регистров способствует увеличению производительности процессора.

Блок *СР* содержит следующие регистры:

старших разрядов *РСР*; используется в командах умножения и деления, а также при переходе от однословного формата к двусловному. Доступен программисту;

базы программы *РБП*, базы данных *РБД1* и *РБД2*, длин сегментов *РДС1* и *РДС0*; предназначены для сегментации областей данных и программ и их защиты. Программист в оперативном режиме имеет возможность сменить сегмент данных. Число таких сегментов памяти, применяемых в программе, ограничивается только объемом ЗУ. Смена нулевого сегмента равноценна смене

решаемых задач. Она возможна только при переходе к подпрограммам и в режиме операционной системы. *РДС0* и *РДС1* фиксируют математические адреса нулевого и первого сегментов;

стека *РСТ*, уровня стека *РУС*; обеспечивают эффективную работу со стеком, причем *РСТ* предназначен для хранения значения верхушки стека, а на *РУС* задается математический адрес последней занятой ячейки памяти, отведенной под стек.

Управляя *РУС*, программист может располагать стек в любом месте сегментов данных, вводить несколько стеков и т. д. Однако в каждый конкретный момент программисту будет доступен только один стек. Введение *РСТ* в процессор позволило ускорить операции со стеком более чем в два раза.

Наличие стека повысило эффективность обращения к подпрограммам, программам обработки прерываний и эффективность реализации программ, записанных на языках высокого уровня;

базы системной таблицы *РБСТ*; предназначен для хранения баз системных процессов.

Блок *БОИ* предназначен для организации асинхронного обмена операндной и командной информацией между процессором, внутренней памятью и устройствами, подключенными к магистрали НЦ.

БОИ содержит:

восемь регистров (*РДП*, *РЧ*, *РКИ*, *РК*, *РАКП*, *РАП*, *РА*, *РСЧК*);

внутреннюю память (*ВП*).

Регистры предназначены для хранения информации, связанной с обменом между процессором и ОЗУ.

РЧ и *РА* — регистры числа и адреса, обеспечивают хранение адреса и данных при обмене с устройствами магистрали НЦ. *РК*, *РКИ* — регистры команд и командной информации используются для приема считанной из ЗУ командной информации и используются для совмещения процессов выборки и выполнения команд процессором. *РСЧК* — регистр-счетчик команд, хранит адрес очередной команды. *РАКП* — регистр адреса конца программы, хранит физический адрес конца программного сегмента. *РДП* и *РАП* — регистры данных и адреса внутренней памяти, используются для временного хранения адреса и данных при обращении к внутренней памяти из магистрали НЦ.

Работа *БОИ* инициируется микропрограммно. При обращении из процессора в *БОИ* передается константа из микрокоманды, которая определяет тип обмена (чтение команды, чтение или запись числа, передача команд управления в магистраль и т. д.). Дальнейшая работа *БОИ* происходит автономно под воздействием сигналов со схемы управления, которая обеспечивает реализацию интерфейса магистрали *НЦ*, интерфейса *ВП* и связь с другими блоками процессора.

При обмене данными физический адрес числа устанавливается в *РА*, а считанное из памяти или записанное в память число поме-

щается в *РЧ*. Вычисление физического адреса и определение принадлежности вычисленного адреса соответствующему сегменту данных, хранящемуся в *ВП*, происходит в *ОБ*. При выходе значения адреса за пределы соответствующего сегмента вырабатывается сигнал прерывания. В начале выполнения программы, а также при переходе в *РСЧК* записывается физический адрес, вычисленный в операционном блоке.

В *ОБ* осуществляется проверка принадлежности адреса сегменту программы. При движении по программе в инкрементном режиме содержимое *РСЧК* сравнивается с содержимым *РАКП*.

При совпадении адресов схема сравнения *СхСр* вырабатывает сигнал прерывания, который по нарушению защиты памяти блокирует обращение в *БОИ* из процессора до конца выполнения текущей микропрограммы.

Адреса данных и команд в процессе обращения к памяти анализируются на принадлежность области адресов *ВП*. Анализ осуществляется сравнением двух старших разрядов *РА* или *РСЧК* с номером процесса. Если адреса входят в область адресов *ВП*, то происходит выполнение операций обмена без выхода в магистраль НЦ. В противном случае происходит захват магистрали.

Регистр *РСП* хранит текущее состояние процессора. Содержит поля признаков результата выполнения команды, режимов процессора и вычислительного процесса, кода прерываний. Поле режимов процессора может устанавливаться как выполняемой программой, так и с магистрали НЦ при выполнении команд в других устройствах или процессорах. Процессор может работать в режимах «Стоп», «Ожидание», «Автомат», «Шаг». Режимы вычислительного процесса определяются признаками: операционной системы (ОС); ловушки отладки (Т); числовой арифметики (Ч); односегментного режима (ОСР).

Так, например, при ОС = 0 запрещено использовать некоторые команды и системные регистры в выполняемых программах.

Признак Т = 1 разрешает переход к операционной системе после команды «Ловушка отладки». Признак Ч = 0 определяет операции с фиксированной запятой. Признак Ч = 1 определяет операции с плавающей запятой.

БМУ обеспечивает выработку последовательности микрокоманд для реализации команд процессора и других специальных функций. Он включает накопитель микрокоманд *НМК*, узел адресации регистров процессора *УАР*, узел формирования адреса микрокоманды *ФАМК*. Объем *НМК* составляет 1024 56-разрядных слов. Каждая микрокоманда имеет две группы независимых полей. Первая содержит адреса операндов и результата, а также управляющие сигналы для различных блоков процессора. Поля второй группы управляют формированием следующего адреса в *ФАМК*.

В узле адресации регистров *УАР* формируются адреса регистров для чтения и записи по соответствующим полям микро-

команды и по содержимому регистра команд *РК*. Доступ к регистрам процессора возможен из микрокоманды, либо из команды.

Адрес микрокоманды формируется в узле формирования адреса микрокоманды. В формировании адреса участвуют: содержимое *РК*, содержимое *РСП* и регистра микрокоманд.

Блок *БПП* принимает сигналы прерываний, хранит их на внутренних регистрах и осуществляет приоритетное их обслуживание в соответствии с микрокомандной информацией.

Сигналы прерываний подразделяют на внешние и внутренние. Обе группы имеют независимое управление. Поступившее в *БПП* прерывание сравнивается с кодом маски, устанавливаемой независимо для каждой группы. При условии, что код прерывания больше кода маски, происходит запись кода прерывания в *РСП*, и из *БПП* в *БМУ* поступает сигнал прерывания, по которому происходит переход к соответствующей микропрограмме. Опрос прерываний осуществляется в конце каждой команды.

Модуль процессора выполнен на базе микросхем серий К155, К589 и КР1802 и размещается на трех платах размером 180×300 мм. Потребляемая мощность 60 Вт. Система команд содержит около 1000 команд с учетом типов адресаций.

Цикл процессора не превосходит 400 нс, что обеспечивает быстродействие процессора 1 млн. операций/с. Процессор может работать с 8-, 16- и 32-разрядными словами, а также с битами и полями битов. В процессоре могут быть обработаны числа с фиксированной и плавающей запятой. При работе в режиме с фиксированной запятой числа представлены в дополнительном коде со знаком в старшем разряде. При работе в режиме с плавающей запятой числа представлены 24-разрядной мантиссой и 8-разрядным порядком. Мантисса и порядок представлены в дополнительном коде.

В целях повышения быстродействия в процессоре применен принцип совмещения выполнения команд на всех уровнях, начиная с совмещения процесса считывания следующей команды с выполнением текущей, считывания операнда из ЗУ совместно с другими действиями, не зависящими от считывания, и заканчивая совмещением процесса выполнения текущей микрокоманды со считыванием следующей.

Высокопроизводительный процессор. Структурная схема процессора (рис. 8.30), реализующего подмножество команд ЕС ЭВМ, содержит следующие блоки и устройства: арифметико-логическое устройство *АЛУ*; блок регистров общего назначения *РОН*; устройство подготовки команд *УПК*; запоминающее устройство микрокоманд *ЗУМК*; устройство прерывания программ *УПП*; блок адресуемых регистров *БАР*; схему синхронизации *СС*; блок сопряжения с ЗУ *БСЗУ*; блок плавающей точки *БПТ*; схему «ω» *Сх*ω.

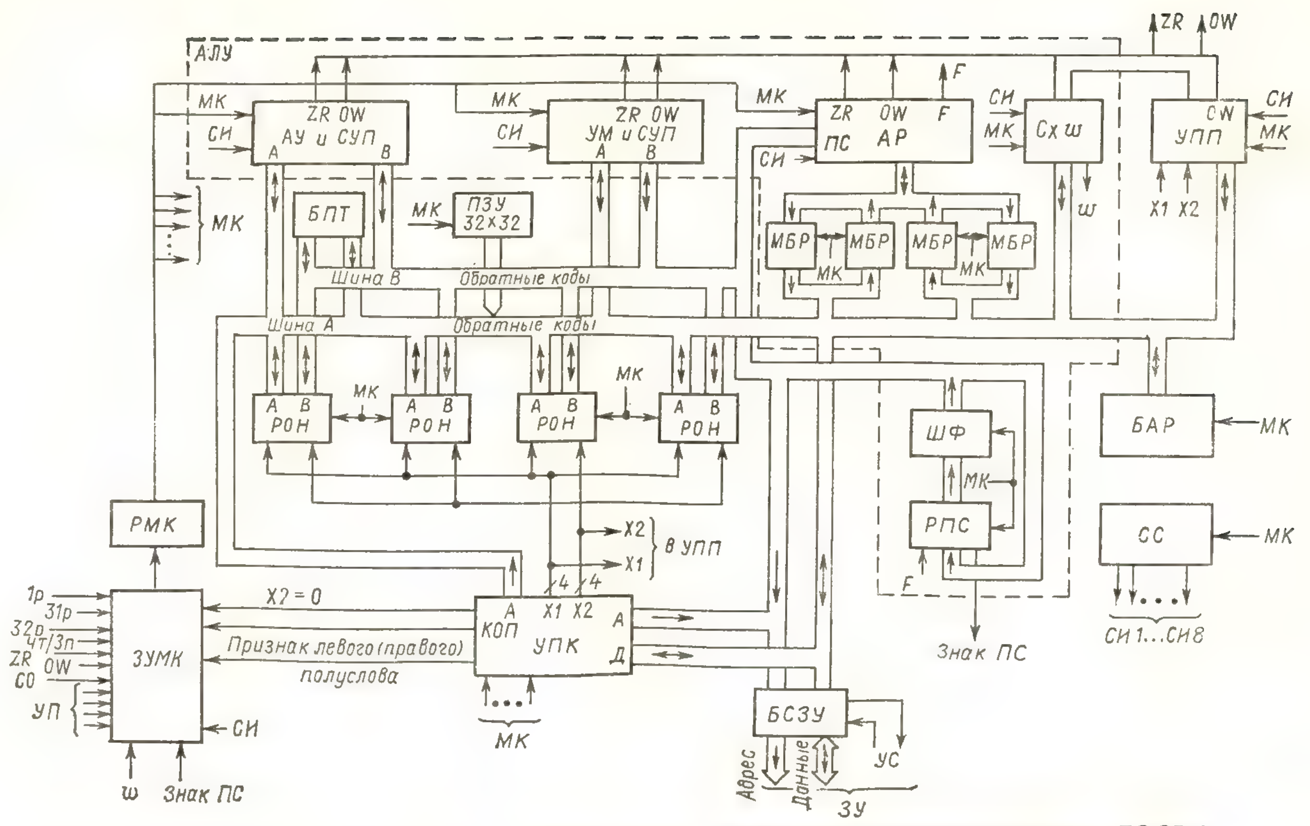


Рис. 8,30. Структурная схема процессора с системой команд, составляющих подмножество команд ЕС ЭВМ

В состав АЛУ входят:

блок *АУ*, предназначенный для реализации арифметических и логических операций над операндами с фиксированной и плавающей точкой (построен на БИС КР1802ВС1);

блок арифметического расширителя *АР*, предназначенный для выполнения операций сдвигов, нормализации, поиска левой единицы (построен на БИС КР1802ВР1);

блок умножения *УМ*, выполняющий операции умножения и деления (построенный на БИС КР1802ВР2).

В *АУ* и *УМ* операнды поступают по магистралям *А* и *В*, по этим же магистралям выдаются результаты операций.

В *АР* входные данные поступают по магистрали *А*, по ней же выдается результат операции. В операциях сдвига параметр сдвига поступает в *АР* из магистрали *В*. Результаты выполнения операций сопровождаются признаками, которые поступают в *ЗУМК*, *УПП* и используются для организации ветвлений на микропрограммном уровне.

АР представляет 16-разрядную БИС КР1802ВР1 без возможности наращивания, позволяющую производить арифметические, логические и циклические сдвиги влево и вправо за один рабочий цикл на 1 ... 15 разрядов, а также поиск левой единицы и операцию нормализации. *АР* позволяет за несколько циклов произвести сдвиг информации разрядностью больше 16 (например, 32, 48, 64). Для подключения 16-разрядного входа (выхода) *АР* к 32-разрядной магистрали используется буферный регистр (построенный на многорежимных буферных регистрах МБР К589ИР12), осуществляющий двунаправленный обмен *АР* с шиной *А*. Регистр параметра сдвига служит для формирования двоичного кода, который содержит величину и направление сдвига для операций сдвига и нормализации, либо номер первого слева разряда, содержащего единицу. *АР* управляется микропрограммно из *ЗУМК*.

Блок умножения состоит из четырех БИС УМН (КР1802ВР2) и схемы ускоренного переноса *СУП*. Блок обменивается с другими устройствами процессора по двум двунаправленным магистралям *А* и *В*. БИС УМН выполняют умножение чисел, деление кодов, операцию загрузки для деления и чтения результатов операций. Для ускорения выполнения операций умножения и деления в блок умножения введена схема ускоренного переноса (К1804ВР1).

Основную функциональную нагрузку в процессоре несет блок *АУ*, который состоит из четырех БИС КР1802ВС1 и *СУП*. БИС КР1802ВС1 выполняют операции конъюнкции, инверсии сложения, дизъюнкции, вычитания, сложения по модулю 2, пересылки, логические, арифметические и циклические сдвиги, операции с внутренним регистром АУ.

Работа блоков *АУ*, *АР* и *УМ* синхронизируется отдельными сериями импульсов, сформированными в блоке синхронизации.

Блок *РОН* предназначен для хранения операндов и результатов выполнения команд. Блок *РОН* емкостью 16 32-разрядных

слов доступен *АУ*, *УМ* и *АР* по записи и чтению в соответствии с микрокомандным словом, поступающим из *ЗУМК*. Обмен информацией с *РОН* происходит по двум двунаправленным шинам *А* и *В*.

Адреса выбираемых регистров задаются из командного слова, хранящегося в *УПК* во время исполнения текущей команды. Блок *РОН* разделен на четыре байта, что обеспечивает доступ к каждому байту, в соответствии с сигналами разрешения из *ЗУМК*. Это используется при выполнении команд с полусловами и байтами.

Устройство подготовки команд *УПК* предназначено для чтения команд из ЗУ, сортировки их по форматам и выдачи кода операции текущей команды в *ЗУМК*, выдачи адресной части команды *А* из *УПК* на шину *А*, выдачи адресов регистров *X1* и *X2* в *РОН*, выдачи сигналов условий в *ЗУМК*. Устройство подготовки команд включает счетчик команд, регистры перекоммутации команд, счетчик групповых пересылок, буферные регистры. *УПК* управляется микропрограммно.

Запоминающее устройство микрокоманд *ЗУМК* предназначено для хранения и выдачи управляющих микрокоманд в блоки процессора. Особенностью работы *ЗУМК* является выборка в одном такте процессора двух последовательных микрокоманд из накопителя с последующей записью их в буферное ЗУ (БЗУ).

В начале следующего такта происходит чтение ячейки БЗУ в регистре микрокоманд, которое занимает меньшее время, чем чтение из накопителя микрокоманд, построенного на ППЗУ.

Адрес ячейки БЗУ модифицируется кодом условия, сформированного в конце предыдущего такта.

В каждом такте процессора происходит перезагрузка БЗУ двумя микрокомандами, одна из которых переписывается в регистр микрокоманд. Это сокращает длительность такта процессора и позволяет осуществлять условные переходы на микропрограммном уровне в следующем такте.

УПП предназначено для обработки прерываний (машинных, программных или внешних), появляющихся в процессе выполнения программы. Причины, приводящие к прерыванию программы, разбиты на группы в порядке уменьшения приоритетов. Допускается прерывание группы с меньшим приоритетом группой с большим приоритетом. Количество причин прерывания — 17, глубина прерываний — 13. *УПП* фиксирует причины прерывания, запоминает состояние машины на момент прерывания и передает управление диспетчерской программе.

УПП построено на базе микросхем серий К589ИК14 с использованием микросхем серий К155 и К531. Реакция на прерывания — 180 нс.

Блок *БАР* предназначен для обмена информацией с шиной *А* и состоит из 16 регистров, адресуемых позиционным кодом. Блок

реализован на микросхемах К589ИР12 и схемах серий К155 и К531.

Блок синхронизации служит для формирования 8 серий синхросигналов и выдачи их в устройства процессора. Включает задающий генератор ($f = 10$ МГц), 8 формирователей синхросерий с периодом $T = 750$ нс и передающих устройств (усилителей и формирователей). Блок выполнен на микросхемах серии К531.

Блок сопряжения с ЗУ предназначен для организации обмена между процессором и ЗУ в режимах записи и чтения. Блок включает двунаправленные магистральные усилители для магистрали данных (шина *А*) и буферный регистр и усилители для магистрали адреса (шина *В*).

Блок реализован на микросхемах К589ИР12 и К531АП2.

Блок *БПТ* предназначен для выполнения вспомогательных функций при реализации на АЛУ группы команд с плавающей точкой. Блок состоит из двух 34-разрядных регистров, каждый из которых разделен на две части (8 и 26 разрядов). Каждая часть имеет раздельное управление. Блок реализован на микросхемах К589ИР12.

Схема «ω» предназначена для формирования двухразрядного признака «ω» результата выполнения команд, построена на схемах ППЗУ К556РТ5.

Выбранная структура позволяет обеспечивать в одном такте (микрокоманде) такие операции как, например, чтение из РОН, арифметико-логические операции с содержимым РОН, запись в РОН результата операции.

Глава 9

РЕКОМЕНДАЦИИ ПО ПРИМЕНЕНИЮ МИКРОПРОЦЕССОРОВ

Применение микропроцессорных БИС ставит перед разработчиками РЭА целый ряд задач, от решения которых зависят технические характеристики разрабатываемой аппаратуры. К таким задачам относятся: выбор оптимального микропроцессорного комплекта БИС, корректная организация линий связи и цепей электропитания, обеспечение заданного уровня помехоустойчивости и обеспечение достоверной работы РЭА, оптимизация конструктивно-технологических решений, автоматизация проектирования программных (микропрограммных) и аппаратных средств, электронное обрамление микропроцессорных средств, вопросы входного контроля, надежности и др.

Основными требованиями к судовой РЭА являются: высокое и сверхвысокое быстродействие основного электронного судового оборудования; большая достоверность полученных результатов; высокая помехоустойчивость [8, 48].

В этой главе рассмотрены вопросы оптимального выбора серии микропроцессорных БИС, организации линий связи, цепей электропитания и экранирования, а также способы контроля получаемых результатов и методика проектирования РЭА на БИС.

9.1. ВЫБОР МИКРОПРОЦЕССОРНОГО КОМПЛЕКТА БИС

В настоящее время отечественная промышленность выпускает свыше 30 серий МПК БИС, объединяющих более 150 типов микросхем средней, большой и сверхбольшой степеней интеграции. Основные вопросы, возникающие на ранних стадиях проектирования микропроцессорной РЭА, сводятся к следующим. Нужен ли микропроцессорный комплект для данной разработки и какой именно МПК БИС наиболее оптимален?

Ответы на эти вопросы в значительной степени определяют технические характеристики РЭА [4, 31, 46, 56].

Использование МПК БИС еще не гарантирует высокие характеристики устройства. Несмотря на положительные качества микропроцессорных БИС, такие как микропрограммируемость (программируемость), широкие функциональные возможности, небольшие масса и габариты, возможность самодиагностики и другие, их внедрение ведет к необходимости менять сами методы и средства проектирования. В первую очередь это приводит к использованию систем автоматизации проектирования, созданию средств поддержки процесса проектирования, переквалификации специалистов, разрабатывающих ранее подобные устройства на «домикропроцессорной» элементной базе.

Основными предпосылками использования МПК при проектировании РЭА являются: высокие требования к габаритным размерам устройств; необходимость перенастраивания на решение других задач; ограниченные сроки разработки.

Первое требование часто сводят к количественным оценкам в микросхемах малой (ИС) и средней (СИС) степеней интеграции, которые нужны для создания подобного устройства. Эти значения колеблются от 50 до 250 корпусов. Иными словами, если МП может заменить 50 ... 250 микросхем малой и средней степеней интеграции в разрабатываемом устройстве, то считается, что его применение оправданно.

Необходимость перенастройки на решение других задач либо на выполнение других команд в случае РЭА, построенной на ИС и СИС по принципу «жесткой логики», приводит к тому, что аппаратуру приходится проектировать заново.

При использовании МП БИС как основы элементной базы данной РЭА смена алгоритма функционирования чаще касается только

запоминающего устройства микрокоманд. Проектирование новых устройств на базе микросхем малой и средней степеней интеграции в основном ведут последовательным способом: вначале создают аппаратную часть устройства, затем — программную.

Возможность одновременного проектирования аппаратной и программной частей микропроцессорной РЭА существенно сокращает сроки разработки. При наличии систем поддержки проектирования выигрыш во времени составляет от 50 до 300 %.

Вместе с тем широкое внедрение МП при разработке устройств выдвигает целый ряд новых требований, которые обусловлены следующими причинами:

отсутствие достаточного числа специалистов по обслуживанию микропроцессорной элементной базы;

отставание выпуска нормативно-технической и справочной документаций от появления новых образцов МП БИС;

необходимость в использовании средств поддержки проектирования различных классов;

изменение традиционных конструктивно-технологических решений;

появление нового этапа проектирования-микропрограммирования.

Таким образом, использование микропроцессорной элементной базы должно быть всесторонне обоснованным.

Если разработчик убежден в целесообразности применения микропроцессора, ему следует выбрать конкретный тип МП. При существующем многообразии типов МПК БИС и их технических характеристик довольно сложно выбрать оптимальный. Существенные различия требований, предъявляемых к судовой РЭА и к решаемым ею задачам, еще более затрудняют этот выбор. Есть целый ряд способов выбора типа МПК. Остановимся на двух наиболее распространенных.

Выбор типа МПК может быть сделан после разработки и анализа бенчмарковских программ. Эти программы представляют собой типовые модели (в несколько сокращенном виде) рабочих задач разрабатываемой РЭА. Обычная длина одной такой программы 100 ... 200 команд [4]. После написания этих программ вычисляется ориентировочное время их выполнения. Для этого программа делится на линейные последовательности команд, имеющие один вход и один выход. Вычисляется время выполнения каждой такой последовательности. Затем из-за наличия условных переходов в бенчмарковской программе вычисляется наилучшее, наихудшее и типовое время выполнения такой программы. За единицу времени выполнения принимают машинный такт РЭА, выполненной на данном МПК. Если время выполнения бенчмарковской программы превосходит требуемое, то идут по пути либо замены типа МПК, либо переложения ряда программных функций на аппаратуру. В случае, если время выполнения бенчмарковской программы существенно меньше требуемого, целесообразнее всего

пойти по пути сокращения аппаратных затрат. Следствием этого будет снижение быстродействия (удлинение машинного такта) и увеличение времени выполнения бенчмарковской программы.

Однако бенчмарковскими программами при выборе МПК пользоваться затруднительно в случае значительной универсальности разрабатываемой РЭА. Наиболее часто используют метод выбора типа МПК на основе количественных эквивалентов характеристик разрабатываемой РЭА. Номенклатура таких количественных эквивалентов (качественных показателей) средств цифровой вычислительной техники (ЦВТ) должна удовлетворять требованиям необходимости и достаточности, т. е. охватывать возможно большее число свойств разрабатываемой продукции для повышения точности оценки и в то же время содержать возможно меньшее число показателей для снижения трудоемкости оценки.

В качестве эталонного, наиболее подходящего к данной разработке, выбирают гипотетический МПК, обладающий лучшими для каждого параметра численными значениями коэффициентов.

Под *качеством* МПК понимается совокупность свойств, обусловливающих его пригодность удовлетворять требованиям технического задания (ТЗ) на аппаратуру.

При выборе типа МПК следует учитывать назначение, область применения, конструктивное исполнение, значения параметров БИС, уровень надежности и ряд других требований ТЗ. Значения параметров БИС влияют на показатели качества МПК. К рекомендуемым параметрам БИС, принимаемым во внимание при выборе серии МПК, следует, в первую очередь, отнести: помехоустойчивость, быстродействие, потребляемую мощность, функциональную полноту, электрическую совместимость с распространенными сериями ИС, число внутренних регистров, разрядность, возможность наращивания, надежность, устойчивость к внешним воздействиям, число номиналов питания.

С учетом конкретных условий работы аппаратуры этот перечень можно сокращать, либо вносить в него новые параметры, представляющие определенную важность для разработчика. Экономические показатели должны вводиться только при оценке функционально и конструктивно законченных изделий.

Конструктивные показатели определяют число выводов БИС данного МПК, а также тип корпуса (*DIP*, Планар и т. д.). При использовании совместно с БИС микросхем малой и средней степеней интеграции возникают ограничения на конструктивное исполнение модулей. Разработчики в основном используют микросхемы в корпусах одного типа, поскольку применение различных типов корпусов отрицательно влияет на общие массогабаритные характеристики аппаратуры и увеличивает число типов модулей.

За помехоустойчивость рекомендуется принимать коэффициент статической помехоустойчивости для уровней «1» и «0» — $K_{п1}$ и $K_{п0}$:

$$K_{п1} = |M_H/U_{THRH}|; \qquad (9.1)$$

$$K_{п0} = |M_L/U_{THRH}|, \qquad (9.2)$$

где M_H и M_L — максимально допустимые уровни напряжений статической помехи для состояний «1» и «0» соответственно; U_{THRH} — минимально допустимый перепад напряжения сигнала из одного состояния в другое.

Из двух значений $K_{п0}$ и $K_{п1}$ выбирают наименьшее.

Мощность потребления определяется для основной БИС комплекта (АЛУ, ЦПУ и т. д.), либо берется максимальное значение мощности потребления для БИС данного МПК. В ряде случаев целесообразно определить мощность потребления для одного разряда обрабатываемой информации. Для БИС, построенных на базе комплементарных МОП-транзисторов, необходимо пользоваться динамическим значением потребляемой мощности для рабочей частоты БИС в аппаратуре.

Быстродействие МПК определяется частотой следования синхроимпульсов, или числом выполняемых операций в единицу времени.

В качестве параметра надежности можно использовать интенсивность отказов (λ), срок службы, ресурс и др. При этом для всех рассматриваемых МПК должны быть выбраны однотипные параметры надежности.

Функциональная полнота МПК определяется наличием в нем разнообразных БИС, позволяющих с высокой эффективностью строить цифровые устройства различного назначения. При определении функциональной полноты (A) рекомендуется учитывать типы БИС или составных частей БИС и их коэффициенты весомости (степени важности для данной разработки):

$$A = \sum_{i=1}^{n} \alpha_i b_i, \qquad (9.3)$$

где n — количество учитываемых типов БИС; i — номер типа БИС; α_i — коэффициент весомости; b_i — множитель наличия БИС либо ее составной части i-го типа («1» либо «0»).

Коэффициенты весомости α для определения функциональной полноты МПК БИС

Тип БИС (составной части БИС)	α
Внутреннее ЗУ (на каждый регистр)	0,05
Регистр сдвига	0,08
Магистральные элементы	0,04
Схемы аппаратного умножения и деления	0,07
Схемы сопряжения с внешними устройствами	0,04
Схемы синхронизаторов	0,01

Схемы управления выборкой следующей микрокоманды	0,12
Схемы организации интерфейсов	0,05
Регистры хранения	0,21
Схемы памяти	0,11
Схемы согласования уровней	0,02
Схемы преобразования информации	0,04
Мультиплексоры	0,06
Схемы приоритетной логики	0,08

Устойчивость к внешним воздействиям определяется в соответствии с техническими условиями на МПК с учетом требований технических условий.

К внешним воздействиям следует отнести механические, климатические, а также специализированные. Условные коэффициенты при механических воздействиях могут колебаться от 0,1 (при вибрации с частотой до 1000 Гц, ускорением до 10g; при многократных ударах с ускорением 10g) до 1,0 (при вибрации с частотой до 10 000 Гц и ускорением до 100 g; при многократных ударах с ускорением 150 g).

Условные коэффициенты в этом случае имеют линейную зависимость, что позволяет определять их промежуточные значения.

Устойчивость к климатическим воздействиям определяется коэффициентами, зависящими от работоспособности БИС при повышенной и пониженной температурах. За единичное значение коэффициента климатических воздействий берется сумма коэффициентов повышенной (α_1) и пониженной (β_1) температур.

Коэффициент α_1 имеет максимальное значение $\alpha_1 = 1{,}0$, если БИС сохраняет работоспособность при температуре $T = 70$ °C. При $T = 35$ °C коэффициент $\alpha_1 = 0{,}14$. Линейная зависимость коэффициента α_1 от температуры позволяет определить его промежуточные значения.

Коэффициент $\beta_1 = 1{,}0$ при $T = -10$ °C; при $T = 0$ °C $\beta_1 = 0{,}1$. Как и для α_1, промежуточные значения коэффициента вычисляются из прямо пропорциональной зависимости.

Устойчивость микросхем к специальным воздействиям определяется в соответствии с конкретными требованиями ТЗ на аппаратуру.

Под электрической совместимостью БИС с распространенными микросхемами понимаются уровни напряжения логических сигналов, а также обеспечение необходимых токов нагрузки. В случае электрической совместимости условный коэффициент равен 1, в противном случае коэффициент равен 0.

Число внутренних регистров определяется, например, коэффициентом, значение которого 1 при числе внутренних регистров 32. При наличии одного регистра в центральной обрабатывающей БИС комплекта коэффициент равен 1/32. Увеличение числа регистров пропорционально увеличивает величину коэффициента.

Разрядность МПК в первую очередь касается центральной обрабатывающей БИС. Коэффициент разрядности может быть равен

1 при разрядности 32 центральной БИС. При разрядности 2 значение коэффициента снижается до 1/16.

Возможность наращивания относится к секционированным МПК. Коэффициент наращивания равен 1 при существующей возможности наращивания. Если такой возможности нет, то он равен нулю. Число номиналов питания также определяется коэффициентом. При одном номинале питания коэффициент равен 1, при двух номиналах — 0,5, при трех — 0,1.

Значения всех параметров нормируются. Нормированный параметр соответствует базовому значению параметра. *Базовым* значением параметра называется величина, принятая за эталон при сравнительной оценке. В качестве базового может быть выбрано наилучшее значение параметра, принадлежащее одному из сравниваемых МПК, или величина, обусловленная требованиями ТЗ.

Нормированные параметры определяются по формуле

$$P = (P_i/P_{i0})^d, \tag{9.4}$$

где P — нормированный параметр; P_i — значение i-го параметра МПК; P_{i0} — базовое значение параметра; $d = 1$, если с возрастанием параметра качество МПК повышается, $d = -1$, если с возрастанием параметра качество МПК снижается.

Единичные показатели качества МПК определяют по каждому параметру в соответствии с формулой

$$q_i = S_i \varphi (P), \tag{9.5}$$

где q_i — единичный показатель качества i-го параметра; S_i — коэффициент весомости i-го параметра; P — нормированный параметр; $\varphi (P)$ — зависимость между нормированным параметром и показателем качества.

Коэффициент весомости S_i характеризует относительную значимость параметра, его влияние на качество МПК и определяется из анализа требований ТЗ на разработку аппаратуры. Кроме этого, в качестве $\varphi (P)$ предлагают следующие зависимости.

При $P \geqslant 1$ рекомендуется пользоваться логарифмической кривой. Здесь

$$q_i = S_i \ln P + 1. \tag{9.6}$$

При $0 \leqslant P \leqslant 1$ рекомендуется пользоваться зависимостью типа

$$q_i = A_i P^3 + B_i P^2 + C_i P + D_i, \tag{9.7}$$

где коэффициенты A, B, C и D определяются из соответствующей нормативно-технической документации.

Комплексный показатель уровня качества Q определяют как среднеарифметическое значение одиночных показателей качества

$$Q = \prod_{i=1}^{n} (\mu/n) \sum_{i=1}^{n} q_i\,, \tag{9.8}$$

где n — число показателей качества; μ — критический множитель, позволяющий обращать в нуль Q в тех случаях, когда какой-нибудь показатель качества оказывается ниже предельно допустимого значения:

$$\mu = \begin{cases} 0 & \text{при} \quad q_i \leqslant 0; \\ 1 & \text{при} \quad q_i > 0. \end{cases} \tag{9.9}$$

Удобно вести расчет показателей качества с помощью нормированных приращений показателей качества:

$$\Delta q_i = q_i - 1, \tag{9.10}$$

т. е.

$$\Delta q_i = \begin{cases} S_i \ln P & \text{при} \quad P \geqslant 1; \\ A_i P^3 + B_i P^2 + C_i P + D_i - 1 & \text{при} \quad 0 \leqslant P < 1. \end{cases} \tag{9.11}$$

В этом случае комплексный показатель качества определяют из равенства

$$Q = \prod_{i=1}^{n} \mu \left(1 + \frac{1}{n} \sum_{i=1}^{n} \Delta q_i\right). \tag{9.12}$$

Суммирование по последней формуле должно быть алгебраическим, с учетом того, что при $q \geqslant 1$ приращение показателей качества положительное, а при $q < 1$ — отрицательное.

МПК с бо́льшим значением Q обладают лучшим качеством для реализации данного ТЗ.

Рассмотрим экономические аспекты использования серий МПК.

Экономический эффект $K_э$ от внедрения МПК можно определить по формулам

$$K_э = K_у + K_{экс} + K_{вр}; \tag{9.13}$$

$$K_у = \Delta C_а + \Delta C_к; \tag{9.14}$$

$$K_{экс} = \Delta C_{э.\,а} + \Delta C_{втр} + \Delta C_{п.\,н}, \tag{9.15}$$

где $K_у$ — экономический эффект от применения МПК, связанный с удешевлением аппаратуры; $K_{экс}$ — экономический эффект от эксплуатации аппаратуры на базе МПК; $K_{вр}$ — экономический эффект от сокращения времени разработки аппаратуры; $\Delta C_а$ — снижение стоимости аппаратуры за счет сокращения числа микросхем, входящих в разрабатываемую РЭА; $\Delta C_к$ — уменьшение стоимости других комплектующих изделий при построении микропроцессорной РЭА; $\Delta C_{э.\,а}$ — экономия собственных амортизационных отчислений; $\Delta C_{втр}$ — экономия затрат на текущий ремонт; $\Delta C_{п.\,н}$ — экономия затрат за счет повышения надежности.

Следует отметить, что экономический эффект от сокращения сроков разработки РЭА на базе МПК достигают за счет распараллеливания процесса проектирования аппаратных и программных средств, а также за счет широкого использования средств автоматизации проектирования.

9.2. ПРОЕКТИРОВАНИЕ РЭА. ОБЩИЙ ПОДХОД

На стадии проектирования функционально сложного быстродействующего устройства, выполненного на больших интегральных схемах, зачастую по целому ряду причин невозможно с достаточной точностью построить подробную временну́ю диаграмму. К таким причинам можно отнести нетривиальные алгоритмы функционирования БИС, недостаток подробной нормативно-технической документации, быстро расширяющуюся номенклатуру БИС и др. Необходимость создания высокопроизводительной судовой аппаратуры на базе быстродействующих комплектов БИС, а также требование «бездефектности» проектирования выдвигают на первый план задачу создания эффективной контрольно-диагностической аппаратуры (КДА), быстродействие которой должно быть значительно выше отлаживаемой РЭА. Создание универсальной КДА вызывает значительные трудности из-за широкого круга задач, решаемых судовой РЭА.

Специализированная КДА и системы автоматизации проектирования (САПР) по трудоемкости и сложности технических решений часто превосходят разрабатываемую аппаратуру, для которой они предназначены. В ряде случаев создание специализированной КДА и САПР нецелесообразно.

Одним из решений этой задачи является выполнение определенных принципов проектирования РЭА, касающихся построения системы синхронизации и логических цепей устройств. Реализация таких принципов делает практически достаточной отладку аппаратуры на пониженных частотах (до 10^5 Гц). В этом случае нет необходимости отлаживать высокопроизводительную РЭА на реальных частотах мегагерцового диапазона. Ниже сформированы основные принципы построения такой РЭА.

1. При разработке РЭА необходимо выполнять требования технических условий и других нормативно-технических документов на используемые электрорадиоизделия. Следует обратить особое внимание на построение узлов синхронизации.

2. При построении схемы устройства необходимо максимально использовать «жесткое» стробирование, при котором элементы последовательной цепи стробируются отдельной последовательностью синхроимпульсов (фаз). Чем больше фазность (число отдельных последовательностей синхроимпульсов), тем выше вероятность бессбойной работы РЭА на рабочих частотах. Максимальное число фаз определяется отношением максимальной задержки из имеющихся во всех ступенях последовательной цепи к минимальной задержке ступени.

3. Следует реализовывать принцип динамического «подобия» процессов, протекающих в РЭА на рабочих и пониженных частотах. Для этого исключать возможность появления «состязаний» входов и «состязаний» цепей, а также импульсных помех. «Состязания» входов возникают из-за неодновременности поступления

сигналов на входные цепи, «состязания» цепей — из-за различия задержек прохождения сигналов по параллельным цепям. Источниками импульсных помех являются электрические неоднородности линий связи, паразитные связи в линиях, импульсные изменения напряжений питания и др.

4. Необходимо использовать принцип одновременной подачи синхроимпульсов, в соответствии с которым синхроимпульсы одной фазы поступают на различные цепи устройства без «перекоса».

После того, как разработаны принципиальная схема и другая схемотехническая документация на РЭА, целесообразно, например, провести масштабное моделирование, т. е. создать макет и отладить его на пониженных частотах, затем выявить ошибки функционального (логического) характера.

Масштабное моделирование может быть проведено на аппаратуре, состоящей из набора контактных устройств, предназначенных для установки в них беспаечным способом корпусов интегральных микросхем. Связь между микросхемами осуществляется штырьковыми соединителями, выполненными в виде витых пар проводов, либо одиночными проводами.

Контактные устройства образуют наборное поле, на котором моделируются наиболее сложные схемотехнические решения. Традиционные решения, общие для подавляющего числа РЭА на БИС, целесообразно вынести за наборное поле, выполняя их в виде отдельного конструктива. Связь между набранным фрагментом и стандартными схемами осуществляется штырьковыми соединителями.

Использование масштабного моделирования и принципов построения РЭА, описанных выше, позволяет выявить практически все ошибки схемотехнического характера, сократив тем самым сроки разработки и регулирования РЭА, а также неэффективную загрузку опытного производства за счет исключения этапа перемонтажа РЭА из-за ошибок разработчика. Кроме того, отпадает необходимость создания дорогостоящей КДА.

9.3. ОРГАНИЗАЦИЯ ЛИНИЙ СВЯЗИ

9.3.1. Общие положения

Среди широкого круга задач, решаемых разработчиками микропроцессорной аппаратуры, важное место занимают вопросы организации линий связи. В подавляющее большинство серий МПК БИС введены специализированные микросхемы, предназначенные для организации линий связи. Надежность работы аппаратуры на базе МПК БИС во многом определяется корректностью организации линий связи на всех уровнях. Линии связи в аппаратуре могут отличаться электрическими параметрами проводных (печатных) соединителей, характеристиками приемников и пере-

датчиков информации, степенью согласования (рассогласования) линии с приемопередающим трактом.

С повышением быстродействия логических элементов скорость обработки информации становится сравнимой со скоростью ее передачи по линиям связи. Это положение справедливо для быстродействующих элементов, построенных на базе ТТЛШ- и ЭСЛ-технологии. Одновременно с ростом быстродействия элементов расширяется частотная полоса пропускания в активной зоне переключения и уменьшается помехоустойчивость элементов. Это особенно заметно при воздействии коротких импульсов, формируемых как сигналами соседних линий, так и внешними помехами. В этих условиях к линиям связи предъявляется ряд требований, выполнение которых влияет на конструкцию проектируемого устройства [10, 49, 53].

Одна и та же линия связи может быть электрически длинной и короткой для разных фронтов сигналов. Условие электрически длинной линии запишется в виде

$$t_0 l \geqslant t_ф, \tag{9.16}$$

где t_0 — погонная задержка в линии, нс/м; l — длина линии, м; $t_ф$ — длительность фронта сигнала, передаваемого по линии, нс.

Погонная задержка линии определяется параметрами линии связи по формуле

$$t_0 = \sqrt{L_0 C_0}, \tag{9.17}$$

где L_0 — погонная индуктивность линии; C_0 — погонная емкость линии.

В случае электрически длинной линии для анализа используют метод распределенных параметров. Для анализа электрически короткой линии связи применяют метод сосредоточенных параметров, заменяя линию эквивалентной схемой. Емкостная и индуктивная составляющие учитываются в схеме генераторами тока и напряжения.

К конструктивным параметрам линии следует отнести вид соединения (проводное или печатное). Если это проводное соединение, то оценивают количество проводов в одной линии (одиночная линия, бифиляр, трифиляр, коаксиальный либо плоский кабели), если соединение выполнено в виде печатных проводников, то параметрами являются геометрические размеры печатных линий связи.

К основным параметрам проводного соединения относят его длину, погонные характеристики, волновое сопротивление.

В табл. 9.1 приведены значения погонной индуктивности, емкости и сопротивления для наиболее часто используемых типов проводов.

К основным характеристикам передатчиков следует отнести выходные токи при «1» и «0», уровни выходных напряжений, фрон-

Таблица 9.1. **Параметры проводных соединений**

Тип провода (проводного соединения)	Волновое сопротивление, Ом	Емкость, пФ/м	Индуктивность, мкГн/м	Сопротивление постоянному току, Ом/м
МНВ (2×0,05 мм²)	103	62	0,66	0,35
МНВ (3×0,05 мм²)	90	67	0,54	0,35
МНВ (1×0,05 мм²)	150 ... 200	≈30	1,0	0,35
ИКМ 0,3/2,4	90 ... 110	46	0,45	0,30
ИКМ 0,12/1,1 ПВП мс	90 ... 110	45	0,46	4,60
75—50×0,2 ПВП мс	67,5 ... 82,5	67	0,38	0,55
100-54×0,18	90 ... 110	52	0,49	0,69
ЛКТ-168-50	120	—	—	0,42
ИКМ-2	100	—	—	0,35
МГДПО-2×0,2	120	—	—	0,20
МГТФ+МС-0,07×0,12	140	66	0,34	0,56
КППРО (20×0,12)	≈90	—	—	0,23
КППР (20×0,2)	≈110	—	—	≈0,15

ты выходных сигналов, выходные сопротивления. В МПК серий К587, К589, К583, К584, К588, К1800, К1802 специально введены микросхемы для организации магистральных линий связи.

Помехи и искажения сигналов в линиях связи возникают при рассогласовании линий на концах, а также из-за взаимного влияния соседних линий друг на друга (перекрестные помехи). Рассмотрим сначала некоторые вопросы, касающиеся рассогласования линий.

Предположим, на одиночную линию работает передающая схема, имеющая внутреннее сопротивление $Z_{\text{вых}}$. На дальнем (приемном) конце расположен приемник с входным сопротивлением $Z_{\text{вх}}$. Волновое сопротивление линии обозначим $Z_{\text{л}}$. Введем

коэффициенты передачи F и отражения на передающем $K_{пер}$ и приемном $K_{пр}$ концах линии связи:

$$\left.\begin{aligned} F = Z_л/(Z_{вых} + Z_л); \quad K_{пер} = (Z_{вых} - Z_л)/(Z_{вых} + Z_л); \\ K_{пр} = (Z_{вх} - Z_л)/(Z_{вх} + Z_л). \end{aligned}\right\} \quad (9.18)$$

Переходные процессы в линии описываются волновым уравнением, которое в операторном виде запишется [49]:

$$d^2\overline{U}_1/dx^2 - p^2\overline{U}_1/V^2 = 0, \qquad (9.19)$$

где $\overline{U}_1$ (P, x) — изображение напряжения в линии; P — оператор Лапласа; V — скорость распространения сигнала в линии.

Изображение сигнала на выходе запишется в виде

$$\overline{U}(P) = A_1(P)\exp(-Pt_л) + A_2(P)\exp(-3Pt_л) + \cdots, \qquad (9.20)$$

где A_1 $(P) = F\overline{U}$ (P) $(1 + K_{пр})$; A_2 $(P) = F\overline{U}(P)(1 + K_{пр})K_{пер}K_{пр}$ и т. д.

При реактивном характере приемников величины F, $K_{пер}$, $K_{пр}$ являются функциями от P. Найдя оригиналы A_i запишем оригинал U_1 (t):

$$\overline{U}_1(t) = A_1(t - t_л)[1(t - t_л)] + A_2(t - 3t_л) + \cdots, \qquad (9.21)$$

где 1 $(t - t_л)$ — единичная функция.

Для активных нагрузок форма сигналов на выходе определяется величиной и знаками коэффициентов отражения. Обозначим длительность фронта сигнала, передаваемого по линии, $t_ф$. Если $K_{пер}K_{пр} < 0$, то процесс на выходе носит колебательный характер. Это можно проверить, проанализировав ряд $\overline{U}(P)$. При $K_{пер}K_{пр} < 0$ ряд знакопеременный. При $K_{пер}K_{пр} > 0$ произойдет искажение фазы передаваемого сигнала. При $K_{пер}K_{пр} = 0$ колебания отсутствуют. Последний случай наиболее благоприятен для передачи сигналов по линии. Это имеет место, например, при равенстве сопротивлений передатчика и линии, тогда формы сигналов в разных точках линии будут подобны друг другу, но различны по амплитуде. Амплитуда сигнала будет уменьшаться от передающего конца линии к приемному, что связано с постоянной составляющей погонного сопротивления линии. Это обстоятельство ограничивает сверху длину согласованной линии связи. Вторым ограничением является запас помехоустойчивости приемников.

Степень рассогласования приемопередающей части и линии при колебательном процессе в тракте передачи характеризуется амплитудой выбросов (помехи). Максимальная амплитуда положительного выброса определяется так

$$U_в = -K_{пер}K_{пр}\Delta U, \qquad (9.22)$$

где ΔU — перепад из «1» в «0».

Положительный выброс может вызвать сбой в приемниках (для ТТЛ и МОП-микросхем). Максимальный отрицательный выброс при колебательном процессе в линии характеризуется следующей амплитудой помехи:

$$U_п = Z_{вх}(K_{пер}K_{пр})^2/(Z_{вх} + Z_{вых}). \quad (9.23)$$

Отрицательный выброс помехи может привести к пробою входных цепей приемника.

Максимальная длина рассогласованной линии связи зависит от $t_ф$ и отношения допустимой помехи к перепаду ΔU ($K_п$):

$$t_ф > 4t_{л0}l_м(1 + K_п), \quad (9.24)$$

где $l_м$ — максимальная длина рассогласованной линии связи.

При $K_п \geqslant 1$ выражение примет вид

$$l_м < t_ф/2t_{л0}. \quad (9.25)$$

Погонная задержка линии связи

$$t_{л0} = \sqrt{L_0C_0}, \quad (9.26)$$

где L_0 и C_0 — погонные индуктивность и емкость линии.

На практике необходимо оперативно оценивать длительность фронтов сигналов, передаваемых по линиям связи.

В работе [10] приведен способ оценки фронтов, основанный на использовании разложения сигнала в ряд. Этот способ достаточно сложный и трудоемкий, он требует применения ЭВМ. Для оценки длительности фронтов сигналов предлагается упрощенный способ, дающий удовлетворительные результаты, пригодные для практической деятельности.

Линия представляется в виде четырехполюсника. Выходной сигнал с четырехполюсника (на входе приемника) описывается выражением

$$U_{вых}(t) = \frac{1}{RC}\int_0^t U_{вх}(t)\,dt. \quad (9.27)$$

Напряжение на выходе передатчика (входе линии) составляет $U_{вх}(t) = K_1t_{ф1}$; $K_1 = U_{вх}/t_{ф1}$; $R = R_{вх} \parallel R_{вых}$; $C = C_{вх} + C_{лин} + C_{монт} + C_{вых}$ при $U_{вх} = 0{,}9U_{вх.мин}$, где $U_{вх.мин}$ — минимально-допустимое входное напряжение «1» для приемника; $t_{ф1}$ — длительность фронта сигнала передатчика в режиме холостого хода.

Тогда

$$t_ф = \sqrt{1{,}8RCU_{вх.мин}/K_1}. \quad (9.28)$$

Указанная формула позволяет оценивать длительность фронтов сигналов, передаваемых по линиям связи длиной до 2,5 м, с точностью не хуже 30 %.

Чаще всего согласование линий связи выполняется тремя способами: одним резистором параллельно на выходе линии; резистивным делителем; последовательно включенным резистором.

В первом случае величина резистора R_0 должна быть равна волновому сопротивлению линии $Z_л$.

Резистор может быть включен между питающей шиной и линией и между линией и земляной шиной источника питания. Это определяется типом передатчиков. В случае ЭСЛ-микросхем согласующие резисторы подключают между линией и отрицательным полюсом источников питания. Параллельное согласование одним резистором относительно низкоомных линий связи приводит к значительным статическим токам, «отбираемым» линией, а это, в свою очередь, значительно увеличивает потребляемую мощность устройства.

Использование резистивного делителя в качестве элемента согласования приводит к понижению статического тока, однако при этом с ростом числа линий связи возрастают габаритные размеры схем согласования.

С одной стороны, номиналы резисторов должны удовлетворять соотношению

$$Z_л = R_1R_2/(R_1 + R_2), \tag{9.29}$$

с другой, номиналы «верхнего» и «нижнего» резисторов должны обеспечивать помехоустойчивые уровни «1» и «0».

С энергетической точки зрения способ согласования с помощью последовательно включенного резистора наиболее экономичен. Действительно, в этом случае к передающей схеме предъявляются минимальные требования по нагрузочной способности. Номинал резистора должен быть равен сопротивлению линии связи минус выходное сопротивление передатчика. Однако нагрузочная способность такого передатчика снижается и становится равной 1—2 нагрузкам для ТТЛ-схем.

Резистор устанавливается последовательно с передатчиком в начале линии. На практике помехи возникают и в согласованных линиях связи, когда они неоднородны по длине. Неоднородности существуют в местах переходов сигналов через разъемы, в точках разветвлений линий и т. п. Согласующие элементы в случае отводов необходимо устанавливать в стыках.

Ниже рассмотрены особенности проектирования линий связи для микросхем, выполненных по различной технологии.

9.3.2. Способы организации линий связи для МОП-микросхем

Выходные каскады МОП-микросхем выполняют в виде стандартных схемотехническо-технологических решений. Наиболее распространенными из них схемами являются: схемы «активный—высокий» и «активный —низкий»; с двумя состояниями; с тремя устойчивыми состояниями; «активный —низкий» с восстановлением «1» и «активный—высокий» с восстановлением «0». Выходы микросхем, не имеющих в оконечных каскадах комплементарных

пар транзисторов, а также выходы, имеющие один активный уровень (с восстановлением), либо три устойчивых состояния, необходимо «доопределять». В случае третьего устойчивого состояния на выходе, «доопределение» необходимо также для ускорения процесса выхода из третьего состояния выходных каскадов микросхем. Если выходы имеют отдельные n-канальные транзисторы с открытым стоком, «доопределение» сводится к подключению нагрузочного резистора к выходу «на питание». Расчетным путем можно получить минимально допустимое $\underline{R}_H$ и максимально допустимое $\overline{R}_H$ значения такого резистора:

$$\underline{R}_H = (\overline{U}_{CC} - \underline{U}_{OL})/I_{OL}, \tag{9.30}$$

где U_{OL} и I_{OL} — выходное напряжение и ток микросхемы при «0» на выходе; $\overline{U}_{CC}$, $\underline{U}_{CC}$ — максимальное и минимальное значения напряжения питания;

$$\overline{R}_H = (\underline{U}_{CC} - U_{OH})/nI_{LIH}, \tag{9.31}$$

где U_{OH} — выходное напряжение при «1» на выходе; I_{LIH} — ток утечки по входу; n — число параллельно подключенных нагрузок.

Номинальное значение R_H должно удовлетворять соотношению

$$\underline{R}_H \leqslant (R_H \pm \% \text{ разброса}) \leqslant \overline{R}_H. \tag{9.32}$$

Если выходные каскады микросхем выполнены на p-канальных транзисторах (с открытым истоком), то «доопределение» производят подключением нагрузочного резистора R_H между выходом микросхемы и землей.

В этом случае минимальное и максимальное значения R_H определяются из уравнений

$$\underline{R}_H = U_{OH}/I_{OH}; \quad \overline{R}_H = U_{OH}/nI_{OH}. \tag{9.33}$$

Номинальное значение R_H выбирают из соотношения $\underline{R}_H \leqslant$ $\leqslant (R_H \pm \%$ разброса$) \leqslant \overline{R}_H$.

Если выходные каскады выполнены по схеме «активный—низкий» и с восстановлением в «1», то значение R_H ($\underline{R}_H$ и $\overline{R}_H$) должно удовлетворять соотношениям (9.30) и (9.31).

Если выходные каскады выполнены по схеме «активный—высокий» и с восстановлением в «0», то номинал R_H определяют как и в предыдущих случаях.

Значения $\underline{R}_H$ и $\overline{R}_H$ можно определить так же, как и в случае отдельных p-канальных транзисторов.

Для организации электрически длинных линий связи для МОП-микросхем следует применять мощные усилительные схемы. Здесь целесообразно использовать, например, микросхему дву-

направленных усилителей К531АП2 (ИС ДНУ), входящую в состав МПК серии К587. Высокое входное сопротивление и большие нагрузочные токи в выходных транзисторах позволяют подключать к выходам усилителей согласованные линии связи длиной в десятки метров. Например, при работе на согласованную линию с волновым сопротивлением 100 Ом и погонным сопротивлением 0,34 Ом/м на приемник с пороговым напряжением $U_{THRL} \geqslant 1{,}4$ В длина линии может достигать 50 м (с учетом запаса помехоустойчивости $M_L = 0{,}4$ В). Открытые коллекторные выходы микросхем К531АП2 позволяют производить ретрансляцию уровней «1» от 2,4 до 10 В, согласуя при этом различные серии микросхем. Эти микросхемы удобны в работе тем, что их можно подключить к выводу резистора, второй вывод которого подключен к напряжению питания 3 ... 10 В. Возможность использования пониженного напряжения питания обеспечивает снижение нижней границы волнового сопротивления согласованной линии без увеличения выходного тока, что выгодно отличает эти микросхемы, например, от шинных формирователей К589АП16, К589АП26, имеющих те же функциональные возможности.

Рассмотрим более подробно способы организации линий связи с использованием микросхем К531АП2. При однонаправленной передаче от ИС ДНУ к МОП-микросхемам согласующий (доопределяющий) резистор R_H подключается между контактом питания и выходом ДНУ. Расчет номинала R_H для всех случаев производится из соотношений (9.30)—(9.32).

При однонаправленной передаче от m ДНУ к n-микросхемам значения $\underline{R}_H$ и $\overline{R}_H$ определяются из выражения

$$\left.\begin{aligned} \underline{R}_H &= (\overline{U}_{CC} - U_{OH})/I_{OL}; \\ \overline{R}_H &= (\underline{U}_{CC} - \underline{U}_{OL})/(nI_{OZ} + mI_{OZH}), \end{aligned}\right\} \qquad (9.34)$$

где U_{CC}, U_{OL}, U_{IH}, I_{OZ}, I_{OL} см. (9.30) — (9.32).

Согласующий резистор устанавливают на приемном конце линии.

Согласование линии связи, выполненной на базе передатчика ДНУ, одним резистором на «Землю» невозможно. Это обусловлено открытым коллекторным выходом ДНУ.

Согласование однонаправленной линии резистивным делителем производят на приемном конце. Значение «верхнего» резистора, подключаемого между напряжением питания и линией, определяется из

$$\underline{R}_{c1} = \overline{U}_{CC}/(I_{OL} - n\overline{I}_{IL}), \qquad (9.35)$$

где n — число приемников.

Значение «нижнего» резистора, подключенного между «Землей» и линией, определяется из условия сохранения уровня «1» на линии

$$\underline{R}_{c2} = \underline{U}_{IH}R_{c1}/(U_{CC} - U_{IH}). \qquad (9.36)$$

Использование согласующего резистора меньше значения R_{c1} может привести к снижению помехоустойчивости ДНУ по уровню «0», либо пробою выходных каскадов ДНУ. Установка «нижнего» резистора, значение которого менее R_{c2}, ведет к понижению помехоустойчивости по уровню «1». Помехоустойчивость по уровню «0» зависит от протяженности линии. Значительные токи, формируемые ДНУ при «0» на выходе, создают падение напряжения на активном сопротивлении линии, снижая помехоустойчивость по «0» M_L при пороговом напряжении «0» U_{THRL}.

Допустимая длина согласованной линии связи $l_{доп}$ для согласования одним резистором

$$l_{доп1} \leqslant \underline{R}_c (U_{THRL} - \overline{U}_{OL} - M_L)/\overline{R}_л (\overline{U}_{CC} - U_{THRL} - M_L); \quad (9.37)$$

для согласования резистивным делителем

$$l_{доп2} \leqslant \underline{R}_{c1} (U_{THRL} - \overline{U}_{OL} - M_L)/\overline{R}_л [\overline{U}_{CC} - (U_{THRL} - M_L) \times \times (1 + \underline{R}_{c1}/\overline{R}_{c2})], \quad (9.38)$$

где $\overline{R}_л$ — максимальное активное сопротивление линии.

Значение минимально допустимого волнового сопротивления согласованной линии связи для передатчиков К531АП2 составляет $R_л = 90$ Ом, напряжение питания $U_{CC} = 3{,}06$ В. Для $R_л = 120$ Ом $U_{CC} = 3{,}7$ В. При более высоких номиналах напряжения источника питания необходимо использовать согласование резистивным делителем.

9.3.3. Организация линий связи для ТТЛ- и ТТЛШ-микросхем

При проектировании линий связи для ТТЛ- и ТТЛШ-микросхем необходимо учитывать их высокое быстродействие (короткие фронты сигналов), значительные выходные и входные токи передатчиков и приемников, заметное различие между входными сопротивлениями приемников при «1» и «0» на входе, относительно невысокую помехоустойчивость [49].

Для обеспечения стабильной работы РЭА на базе ТТЛ-микросхем применяют уже рассмотренные ранее способы согласования. Приведем расчет согласующих элементов, удовлетворяющих условиям согласования, ограничениям на выходные токи передатчиков и обеспечению помехоустойчивости работы приемопередающего тракта.

При согласовании линии одним резистором на «Землю» минимально допустимое значение согласующего резистора определяется из условия ограничения на выходной ток «1»:

$$R_C = U_{OH}/I_{OH}.$$

При типовых значениях выходных токов по «1» микросхем серии К155 номинальное значение R_c составит 5 ... 6 кОм. Для передатчика типа К589АП16 (АП26) R_c уменьшится до 480 Ом.

Анализируя линии передачи, приведенные в табл. 9.1, можно прийти к выводу: для ТТЛ- и ТТЛШ-микросхем наиболее распространенные линии связи не могут быть согласованы резистором на «Землю».

Для способа согласования линии связи одним резистором на «Питание» значение согласующего резистора можно определить из ограничения по выходному току передатчика при «0»:

$$\bar{I}_{OL} \geqslant (\bar{U}_{CC} - \underline{U}_{OL})/\underline{R}_c + n I_{IL}. \qquad (9.39)$$

При выполнении условий $n\bar{I}_{IL} \ll \bar{I}_{OL}$ и $U_{OL} \approx 0$ получим ограничение на минимальное значение согласующего резистора:

$$\underline{R}_c \geqslant \bar{U}_{CC}/\bar{I}_{OL}. \qquad (9.40)$$

Для большинства ТТЛ- и ТТЛШ-микросхем минимальное значение R_c = 330 ... 230 Ом. Для передатчиков К589АП16 (АП26) минимальное значение R_c с учетом разброса составит 130 Ом.

Таким образом, резистором на «Питание» можно согласовывать однонаправленные линии связи с волновым сопротивлением ⩾130 Ом для передатчиков К589АП16 (АП26).

Длина согласованной линии связи ограничится в этом случае количеством приемников, активной составляющей волнового сопротивления, а также пороговым напряжением и запасом помехоустойчивости по уровню «0»:

$$l \leqslant \frac{U_{THRL} + M_L - U_{OL}}{R_л \left[n\bar{I}_{IL} + \frac{\bar{U}_{CC} - (U_{THRL} + M_L)}{R_C} \right]} \qquad (9.41)$$

(при этом рассматривался наихудший случай — все приемники подключены к концу линии).

Ограничение на количество приемников при обеспечении помехоустойчивости по уровню «1» запишется в виде

$$n \leqslant \frac{\underline{U}_{CC} - (U_{THRH} + M_H)}{\bar{R}_C \bar{I}_{IH}}. \qquad (9.42)$$

Здесь также имелся в виду наихудший случай сочетания параметров и предполагалось, что все приемники подключены к концу линии.

Для микросхем К589АП16 (АП26) пороговые значения входных напряжений «0» (U_{THRL}) и «1» U_{THRH} составляют 0,8 и 2 В.

Для передатчиков типа К589АП16 (26) $R_л$ = 0,4 Ом/м, волновом сопротивлении линии 130 Ом, $\bar{U}_{CC}$ = 5,25 В, n = 1 шт. длина согласованной линии связи составит 3 м. При этом запас помехоустойчивости по уровню «0» составит 0,4 В.

Если в качестве согласующего элемента использовать резистивный делитель, то условие электромагнитного согласования запишется в виде

$$R_{c1}R_{c2}/(R_{c1} + R_{c2}) = Z_л, \qquad (9.43)$$

где $Z_л$ — волновое сопротивление линии.
Следует выполнять соотношение

$$\overline{I}_{OL} \geqslant \frac{\overline{U}_{CC} - \underline{U}_{OL}}{\underline{R}_{c1}} + n\overline{I}_{IL} - \underline{U}_{OL}/\overline{R}_{c2}. \qquad (9.44)$$

Сделав допущения $U_{OL} = 0$ В, $n\overline{I}_{IL} < \overline{I}_{OL}$, ограничение на величину R_{c1} находят из следующей формулы:

$$R_{c1} \geqslant \overline{U}_{CC}/\overline{I}_{OL}. \qquad (9.45)$$

При «0» на выходе передатчика допустимое число приемников может быть определено по формуле

$$n \leqslant \frac{1}{\overline{I}_{IH}\overline{R}_{c1}} \left[\underline{U}_{CC} - \frac{(U_{THRH} + M_H)(\overline{R}_{c1} + \underline{R}_{c2})}{\underline{R}_{c2}} \right]. \qquad (9.46)$$

Ограничение на длину линии согласованной делителем запишется из условия обеспечения помехоустойчивости по уровню «0»

$$l \leqslant \frac{U_{THRL} + M_L - U_{OL}}{R_л \left[\frac{\overline{U}_{CC} - (U_{THRL} + M_L)}{R_{c1}} + n\overline{I}_{IL} - \frac{U_{THRL} + M_L}{R_{c2}} \right]}. \qquad (9.47)$$

Следует отметить, что расчет минимально допустимого значения резистора R_{c2} выполняют аналогично расчету R_c для согласования резистором на «Землю»

$$R_{c2} \geqslant U_{OH}/I_{OH}. \qquad (9.48)$$

Таким образом, после несложных расчетов получим минимальное значение волнового сопротивления однонаправленной линии, согласованной делителем для передатчиков К589АП16 (26). Оно составит 110 Ом. Максимальная длина линии связи в этом случае при $n = 1$ и удельном сопротивлении = 0,4 Ом/м будет 15 ... 20 м.

Выше рассматривались однонаправленные линии связи. При организации двунаправленных линий согласующие элементы необходимо устанавливать на обоих концах линии, а способ согласования резистором на «Землю» не применим.

При согласовании резистором на «Питание», учитывая, что оба резистора на двух концах линии включены параллельно, получим ограничение на минимально допустимое значение одного R_c:

$$\underline{R}_c = 2\overline{U}_{CC}/\overline{I}^0_{OL}. \qquad (9.49)$$

Таким способом нельзя согласовать наиболее распространенные линии связи с использованием приемопередатчиков типа К589АП16 (26). При согласовании двунаправленной линии рези-

стивным делителем ограничение по выходному току «0» передатчика запишется в виде

$$\overline{I}_{OL} \geqslant 2\frac{\overline{U}_{CC}-U_{OL}}{\underline{R}_{c1}}+n\overline{I}_{IL}+(n-1)I_{OZ}-\frac{2U_{OL}}{\underline{R}_{c2}}. \qquad (9.50)$$

Сделав допущения как в случае согласования однонаправленной линии, получим:

$$\underline{R}_{c1} \geqslant 2\overline{U}_{CC}/\overline{I}_{OL}. \qquad (9.51)$$

Минимально допустимое значение R_{c2} можно найти, анализируя ограничение на выходной ток передатчика при «1»

$$\underline{R}_{c2} = 2\underline{U}_{OH}/I_{OH}. \qquad (9.52)$$

Для передатчиков К589АП16 (26) минимальные значения R_{c1} и R_{c2} составят: R_{c1} = 270 Ом; R_{c2} = 1200 Ом.

Таким образом, двунаправленные линии связи, организованные приемопередатчиками К589АП16 (26) с волновым сопротивлением $Z_л \geqslant$ 220 Ом, можно согласовывать резистивным делителем. Следовательно, чем выше выходной ток передатчика и ниже напряжение источника питания, тем шире возможности для согласований различных линий. Приемопередатчики, входящие в МПК серии К589, имеют явно недостаточные возможности для организации двунаправленных проводных линий связи. Поэтому следует обратить внимание разработчиков на микросхемы К531АП2, К559ИП1 ... 3 и другие, имеющие более высокие нагрузочные возможности.

9.3.4. Перекрестные помехи в линиях связи

До сих пор рассматривались одиночные линии связи. Однако на практике чаще приходится иметь дело со жгутовыми линиями связи, в которых в непосредственной близости друг от друга расположены две и более сигнальные линии. В каждый момент времени активизируются передаваемыми сигналами не все линии. Некоторые из них остаются невозбужденными (пассивными). При наличии взаимной связи между линиями, сигналы активных линий наводятся в пассивных. Чем больше длина участков, имеющих значительную взаимную связь, тем большее напряжение сигнала помехи наводится в пассивных линиях. Такие помехи имеют индуктивную и емкостную составляющие. Соотношение между ними в печатной плате без металлизированных поверхностей выражается примерной формулой

$$U_c/U_л \approx (C_{B0}/L_{B0})(R_O/N_a), \qquad (9.53)$$

где U_c и U_L — амплитуды емкостной и индуктивной составляющих помехи; C_{B0} и L_{B0} — взаимные емкость и индуктивность линий;

R_O — выходное сопротивление передатчика пассивной линии; $N_а$ —число нагрузок в активной линии [7, 10, 49, 54].

Отсюда следует, что соотношение между амплитудами емкостной и индуктивной составляющих помехи зависит только от числа нагрузок в активной линии и выходного сопротивления управляющего элемента в пассивной. В печатной плате без металлизированных поверхностей ограничение на длину взаимодействующих проводников запишется в виде

$$l \leqslant \frac{M_L t_ф 10^3}{\sqrt{\varepsilon}\left[U_а\left(33{,}3\frac{R}{Z_л}\right)\right] + \frac{R_I Z_л N_а I_I}{15\,(R_I + R_O)}}, \quad (9.54)$$

где M_L — допустимое напряжение статической помехи, В; $t_ф$ — длительность фронта передаваемого сигнала, нс; ε — относительная диэлектрическая проницаемость; $U_а$ — амплитуда активного сигнала в линии, В; $Z_л$ — волновое сопротивление линии, Ом; $R = R_I \parallel R_O$, Ом; I_{IL} — входной ток «0».

При взаимодействии проводника в печатной плате с большим количеством близко расположенных проводников должно выполняться неравенство

$$|M_L| \geqslant C_0 R U_а / t_ф, \quad (9.55)$$

где $|M_L|$ — абсолютная величина запаса помехоустойчивости приемников линии, В; C_0 — суммарная емкость между выбранным проводником и остальными проводниками на плате, Ф.

С точностью, достаточной для практических расчетов, суммарную емкость C_0 заменяют емкостью между печатным проводником и проводящей плоскостью платы (модуля). $U_а$ и $t_ф$ — как и для соотношения (9.54).

Предельная длина проводников, имеющих значительную взаимную связь в плате без металлизированной плоскости, составляет

$$l_п \leqslant M_L t_ф Z_л C_0 \sqrt{\varepsilon} R U_а. \quad (9.56)$$

Это означает, что при любом размере печатной платы и произвольном расположении проводников амплитуда помехи не превышает допустимый уровень при длине проводника $l_п$, удовлетворяющей неравенству (9.56). Введение в печатную плату металлизированных поверхностей «Земля» и «Питание» позволяет существенно снизить взаимное влияние линий связи, а значит, и уменьшить уровень перекрестных помех. При переключении передатчика активной линии из «1» в «0» особое значение (до 75 %) имеет емкостная составляющая помехи. В печатных платах с металлизированными поверхностями снижение уровня перекрестных помех в пассивной линии можно определить из выражения

$$F = \frac{x_2}{x_1}\,\frac{1 - e^{(-1/x_2)}}{1 - e^{\,1 + \frac{1}{x_1(1 + C_л/C_{св})}}}; \quad (9.57)$$

$$x_1 = RC_{св1}/t_ф; \quad (9.58)$$

$$x_2 = RC_{св2}/t_ф, \quad (9.59)$$

где $R = (R_I R_0)/(R_I + R_0)$, Ом; $C_{св1}$ — емкость связи между пассивными линиями в плате с металлизированными поверхностями; $C_{св2}$ — емкость связи между пассивными линиями в плате без металлизированных поверхностей, Ф; $C_л$ — собственная емкость пассивной линии, Ф; $t_ф$ — длительность фронта сигнала, передаваемого по активной линии, с. Для навесного объемного монтажа, выполненного отдельными проводниками, лежащими в одной плоскости и проложенными по кратчайшему расстоянию методом «Путанка», можно записать неравенство

$$|l_п| \leqslant \frac{1{,}79 M_L t_ф}{U_\alpha R \varepsilon} \ln\left[\frac{h}{r}\left(1 + \sqrt{1 - \frac{d_1^2}{d_2^2}}\right)\right], \quad (9.60)$$

где h — расстояние от плоскости платы до выбранного проводника, м; d_1 и d_2 — внутренний и внешний диаметры проводника в изоляции, м; ε — относительная диэлектрическая проницаемость изоляции.

Для жгутового соединения предельная длина составит

$$|l_п| \leqslant \frac{1{,}79 M_L t_ф}{U_a R \varepsilon} \ln \frac{4h + d_1}{d_1}. \quad (9.61)$$

9.4. ОРГАНИЗАЦИЯ ЦЕПЕЙ ПИТАНИЯ

От корректной организации цепей питания зависит надежность работы проектируемого устройства. Для МПК серий К583 и К584, построенных на базе $И^2Л$-технологии, при организации цепей питания необходимо использовать источники тока (а не напряжения). От правильности выбора элементов для цепей питания зависит быстродействие устройств на этих микросхемах.

Токи, протекающие по цепям питания и напряжения, которые они развивают, зачастую могут явиться источниками помех и сбоев цифровых устройств. В случае МОП-микросхем, в особенности высокопороговых МПК, где при переключении отдельных микросхем токи изменяются на 50 ... 1000 мкА, решение проблемы сохранения заданной помехоустойчивости РЭА не вызывает затруднений. Однако при увеличении энергетических характеристик серий МПК эта проблема актуальна. Так, для элементов ТТЛШ типичен выброс тока по цепи питания при переключении вентиля из «0» в «1». Длительность его составит 7 ... 12 нс при амплитуде 10 ... 200 мА. Образовавшийся выброс прикладывается по цепям питания к соседним элементам, вызывая их ложные срабатывания. При наличии индуктивной связи цепей питания с сигнальными линиями подобные выбросы могут вызвать дополнительные помехи [53, 58].

Шины питания на платах обычно прокладываются печатными проводниками. Практическое значение волнового сопротивления таких проводников $Z_л \leqslant 20$ Ом.

Основными задачами конструирования цепей питания являются снижение влияния погонной индуктивности в пределах платы и обеспечение периодического характера установления напряжения питания при переключениях нагрузки. Кроме печатного монтажа применяется навесной. При этом предпочтительнее изготовлять шины «Питания» и «Земли» смежными при толщине шины не менее 4 мм и толщине разделительного диэлектрика 0,05 ... 0,15 мм.

Для обеспечения периодического характера изменения напряжения питания каждое звено микросхем на плате, представляющее собой последнюю ступень в схеме разветвления питания, должно содержать персональную группу конденсаторов *C1* (электролитического) и *C2* (керамического), подключаемых между шинами «Питания» и «Земли». Емкости этих конденсаторов из расчета на одну микросхему должны быть не менее 0,1 мкФ для *C1* и 0,002 для *C2*. В качестве конденсатора *C2* рекомендуется применять конденсаторные сборки Б18, керамические конденсаторы К10-17-2а-Н90 — 0,033 мкФ, либо К10-23 — 0,033 мкФ, КМ5, КМ6.

В качестве конденсатора *C1* можно применять К53-4а-16В — 3,3 мкФ $\pm$ 20 %.

Конденсаторные сборки следует располагать по углам платы, керамические конденсаторы типа К10-23 — посередине платы вдоль поперечных шин «Питания». Электрические конденсаторы рекомендуется располагать на вводе шин питания в плату. Микросхемы большой степени интеграции (ТТЛШ-, И²Л-, ЭСЛ-технологий) следует устанавливать как можно ближе к месту ввода питания на плату.

Использование фильтрующих конденсаторов позволяет сгладить влияние пульсаций в шинах питания в n раз, при этом

$$n = U_п/U_{пф} = \sqrt{L_0C_0}/2t_ф, \quad (9.62)$$

где $U_п$ — помеха по цепи питания без фильтрующих конденсаторов; $U_{п.\,ф}$ — помеха по цепи питания при наличии фильтрующих конденсаторов; C_0 — суммарная емкость фильтра на одно звено (5 ... 10 микросхем); L_0 — погонная индуктивность шины питания (обычно $L_0 = 0,02$ мкГн/см); $t_ф$ — длительность фронта переключения сигнала.

Межблочные шины питания не всегда удается выполнить в виде печатных проводников. При проектировании межблочных цепей питания для ТТЛШ- и И²Л-микросхем рекомендуется использовать провода типов МГТФ, МПКМ, МГШВ. Для уменьшения индуктивности этих цепей в 3—4 раза разводку питания следует вести витой парой проводов.

Например, провод диаметром $d_1 = 0,5$ мм, с изоляцией толщиной $a = 0,5$ мм и длиной $l = 0,5$ м имеет индуктивность

≈730 нГн. Проложив этот участок свитой парой проводов, получим индуктивность 230 нГн. При скрутке следует стремиться к уменьшению расстояния между проводами. Индуктивность свитой пары проводов определяется как

$$L_{св} = 2l \ln \frac{2(d_1 + 2a)}{d}, \qquad (9.63)$$

индуктивность одиночного провода —

$$L = 2l \ln [4l/(d_1 - 1)]. \qquad (9.64)$$

При питании устройства от нескольких источников их общие выводы объединяются по кратчайшему пути в общую точку. В целях уменьшения влияния изменений номинала одного напряжения питания в первичной цепи на выходное напряжение источников питания вводят стабилизаторы сетевого напряжения.

Для уменьшения негативного влияния импульсной нагрузки в стабилизированных источниках питания вводят отрицательную обратную связь проводами малых диаметров. Между проводами обратной связи и силовыми проводами одинаковой полярности должны быть подключены безындукционные керамические конденсаторы (например, типа КМ6) емкостью 1 мкФ.

Для сглаживания бросков по первичной сети (если нет возможности установить стабилизаторы) необходимо устанавливать конденсаторные фильтры. Емкость конденсатора выбирают из расчета 1200 мкФ на 1 А тока нагрузки вторичного источника питания.

9.5. ЭКРАНИРОВАНИЕ ОТ НИЗКОЧАСТОТНЫХ ПОЛЕЙ

Источниками низкочастотных магнитных полей чаще всего являются силовые трансформаторы цепей питания. Для защиты от низкочастотных полей применяют магнитостатический экран из ферромагнита. Обычно магнитостатические экраны выполняются из трансформаторной стали со стенками толщиной 2 мм. Существующие швы и окна в материале экрана не должны располагаться поперек ожидаемого направления магнитных линий. Магнитостатический экран можно не заземлять. Эффективность экранирования повышается при двойном экране. В этом случае коэффициент экранирования выше, чем при использовании толстостенных экранов. Коэффициент экранирования можно оценить по формуле

$$K_э = B_1/B_0 = 1 + 0{,}66b\mu/R_0, \qquad (9.65)$$

где B_1 — индукция внутри экрана; B_0 — индукция внешнего магнитного поля; b — толщина стенки экрана; μ — магнитная проницаемость материала экрана; R_0 — радиус сферы, равновеликой (по объему) экрану.

Если цифровая аппаратура работает в условиях мощного электростатического поля, то прибегают к экранированию аппа-

ратуры. В качестве материала для экрана используют металлы, имеющие хорошую электрическую проводимость: медь, серебро, алюминий. Электростатический экран в отличие от магнитостатического обязательно заземляют.

Защита аппаратуры от высокочастотного внешнего электромагнитного поля заключается в том, что при прохождении через металлический лист вследствие наличия вихревых токов это поле значительно затухает. В частности, длина волны λ в проводящей среде металлического экрана в виде короба вычисляется по формуле

$$\lambda = 2\sqrt{\pi}/\sqrt{f\mu\mu_0\gamma}, \tag{9.66}$$

где f — частота изменения электромагнитного поля; μ, μ_0 — магнитные проницаемости металла на данной частоте; γ — удельная проводимость металла.

На расстоянии $\lambda/2$ вглубь от поверхности экрана энергия внешнего поля практически затухает. Следовательно, толщина экрана при выбранном материале должна иметь значение, близкое к λ. В общем случае для защиты цифровых устройств от внешних полей с широким спектром частот экран следует выполнять комбинированным (двойным) из электротехнической либо трансформаторной стали с $\mu \leqslant 200$ и из металла с высокой удельной проводимостью (медь, серебро). Металл часто наносят в виде покрытия. Иногда достаточно установить комбинированный экран из алюминия и стали. В качестве такого экрана можно использовать шасси блоков и металлическую обшивку стоек [53, 54].

Описанные выше методы экранирования относят к защите цифрового устройства от мощных внешних полей. Однако не менее важной задачей является экранирование от соседних источников полей в цифровом устройстве. Экраном в этом случае могут служить оплетки проводов или же так называемые «кипера». Этот способ заключается во введении в конструкцию плоского магнитовода, примыкающего непосредственно к плоскости разводки управляющих проводников, что приводит к значительному снижению емкостной и индуктивной связей между проводниками.

9.6. ОРГАНИЗАЦИЯ КОНТРОЛЯ ЦИФРОВОЙ МИКРОПРОЦЕССОРНОЙ АППАРАТУРЫ

При внедрении микропроцессоров в РЭА необходимы жесткие требования к средствам ее контроля. Микропроцессорные БИС, являясь, по существу, сложными автономными устройствами, проверяются теми же методами, что и модули на микросхемах малой и средней степеней интеграции. В настоящее время микропроцессорная аппаратура проверяется так же, как и миниЭВМ. Не касаясь методов входного контроля, которые сами по себе являются темой для работы значительного объема, остановимся на методах контроля выполнения операций в вычислительных устройствах.

Требования к надежности и достоверности результатов, полученных вычислительным устройством, постоянно возрастают. Повышение надежности и достоверности работы РЭА решают несколькими путями: повышением надежности узлов и деталей; введением в аппаратуру структурной избыточности; применением контроля устройств.

Методы контроля делят на программные, аппаратные и комбинированные [21, 22, 40].

Программные методы контроля включают:

программно-логический контроль, реализуемый двойным просчетом с последующим сравнением результатов;

алгоритмический, заключающийся в выполнении поставленной задачи по более короткому алгоритму;

тестовый контроль, состоящий в выполнении тестовых последовательностей с последующим сравнением результатов с заданными.

В настоящее время наиболее широко используют аппаратный и комбинированный способы контроля.

Аппаратный контроль заключается во внесении в разрабатываемую РЭА схем, которые реализуют контроль выполнения операций. Такой контроль выполняется, как правило, с минимальными затратами времени в каждом машинном цикле. Аппаратный контроль бывает в основном трех видов, а именно: контроль по модулю; дублирование, троирование вычислительных средств; контроль избыточными кодами.

Комбинированный контроль заключается в совместном использовании первых двух видов контроля. Например, перед началом решения задачи вычислительное устройство выполняет тестовые проверочные программы с целью выявить неисправности или, если таковые отсутствуют, проинформировать о готовности данного устройства проводить вычисления. В этом случае аппаратный контроль заключается в проверке вычислений (контроль по модулю) в машинных тактах, либо при выполнении определенных операций.

При контроле обычно выявляются случайные или систематические машинные сбои. Случайные сбои не зависят от характера перерабатываемой информации и определяются случайными воздействиями, такими как электромагнитные помехи, температурные и климатические изменения и т. д. Например, при контроле методом двойного просчета сбой длительностью $\tau_с$ будет случайным, если

$$\tau_с < T_ц/2, \tag{9.67}$$

где $T_ц$ — время двойного просчета.

При $\tau_с > T_ц$ сбой считается систематическим.

Основное отличие случайных сбоев от систематических в том, что их нельзя прогнозировать. К систематическим сбоям относят сбои «залипшая» единица, «залипший» нуль. Они могут появ-

ляться при старении элементов, ухудшении электрических контактов, неправильной эксплуатации устройства, выхода за допустимые пределы напряжений источников питания.

С математической точки зрения сбои характеризуются вероятностью появления за время θ, а с точки зрения пользователя — временем безотказной работы.

Так, вероятность появления ошибки в n-разрядном двоичном коде определяется из выражения

$$P_i = \frac{n\mu_{\text{р}} t_{\text{оп}}}{i!} e^{-n\mu_{\text{р}} t_{\text{оп}}}, \qquad (9.68)$$

где n — число разрядов кода; $\mu_{\text{р}}$ — интенсивность сбоя в одном разряде устройства; $t_{\text{оп}}$ — время выполнения операции в устройстве; i — число ошибок (однократная, двойная, тройная и т. д.).

С увеличением i уменьшается вероятность появления i-кратной ошибки. При $i = 4$ вероятность появления такой ошибки пренебрежимо мала. Поэтому при рассмотрении эффективности методов контроля ограничиваются значениями $i = 1, 2, 3$.

При использовании контроля по модулю однократная ошибка обнаруживается с вероятностью $P = 1$.

Вероятность пропуска двукратной и трехкратной ошибки при контроле по модулю 3 находится из выражения

$$P_{2,\,3} = 0{,}25 n^2 \mu_{\text{р}}^2 t_{\text{оп}}^2 (1 + 0{,}166 n\mu_{\text{р}} t_{\text{оп}}). \qquad (9.69)$$

Интенсивность сбоя всего устройства μ определяется из выражения

$$\mu \approx \xi\mu_{\text{р}}, \qquad (9.70)$$

где

$$\xi = Q_{\text{р}}/Q_{\text{о}}; \qquad (9.71)$$

$Q_{\text{р}}$ — количество единиц оборудования одного разряда; $Q_{\text{о}}$ — количество единиц оборудования всего устройства. Например, при $\mu = 0{,}1$ 1/ч $\xi = 0{,}001$, $\mu_{\text{р}} = 10^{-4}$ 1/ч.

При контроле методом дублирования пропуск ошибки произойдет лишь при условии одновременного одинакового сбоя в одноименных разрядах основного и дублирующего устройств.

Вероятность появления ошибки в двух одинаковых разрядах основного и дублирующего устройства

$$P_1' = \frac{n^2 \mu_{\text{р}}^2 t_{\text{ср}}^2}{2} e^{-2n\mu_{\text{р}} t_{\text{ср}}} \frac{1}{n^2}. \qquad (9.72)$$

Вероятность появления двухкратной ошибки

$$P_2' = \frac{\mu_{\text{р}}^4 t_{\text{ср}}^4}{(2!)^2} e^{-2n\mu_{\text{р}} t_{\text{ср}}}. \qquad (9.73)$$

Вероятность появления трехкратной ошибки

$$P_3' = \frac{\mu_{\text{р}}^6 t_{\text{ср}}^6}{(3!)^2} e^{-2n\mu_{\text{р}} t_{\text{ср}}}. \qquad (9.74)$$

Общая вероятность появления одно-, двух- и трехкратных ошибок при дублировании запишется в виде

$$P_о = P'_1 + P'_2 + P'_3, \tag{9.75}$$

т. е.

$$P_о = \mu_р^2 t_{ср}^2 e^{-2n\mu_р t_{ср}} \left(1 + \frac{\mu_р^2 t_{ср}^2}{4} + \frac{\mu_р^4 t_{ср}^4}{36}\right). \tag{9.76}$$

Вероятность пропуска ошибок для пары одноименных разрядов можно вычислить по следующей упрощенной формуле:

$$P'_{2п} \approx \mu^2 t_{ср}^2. \tag{9.77}$$

На практике бывает важно знать вероятность появления ошибок (сбоев) за определенное время θ. Так, вероятность появления сбоев в устройстве без контроля определится из выражения

$$P_д(\theta) = e^{-\mu\theta}. \tag{9.78}$$

Вероятность бессбойной работы устройства за время θ при контроле по модулю определится так:

$$P_{д.\,м}(\theta) \approx e^{-\varepsilon\mu\theta}, \tag{9.79}$$

где $\varepsilon = \mu'/\mu$; μ' — интенсивность сбоев в оборудовании, не охваченном контролем; μ — интенсивность сбоев во всем устройстве.

При дублировании резко повышается вероятность бессбойной работы устройства. В этом случае сбой будет замечен на самых ранних этапах его появления и локализован.

При троировании устройства с применением методов мажорирования двух из трех эффективность контроля сравнима с эффективностью контроля при дублировании. Однако этот метод позволяет не только осуществлять получение достоверных результатов, но и обнаруживать неисправную часть (одну из трех) системы в целом. Этим методом пользуются, в частности, при выполнении ответственных штурманских вычислений, для решения задач ориентирования, в центральных системах управления движением судна.

Приведем вероятность безотказной работы управляющего устройства за время θ. Вероятность безотказной работы за время θ устройства без контроля

$$P_0(\theta) = e^{-\left(1 + \frac{\mu}{\lambda}\right)\theta}, \tag{9.80}$$

где λ — интенсивность отказов устройства;

в устройстве с контролем по модулю

$$P_м(\theta) = e^{-\lambda\,(1+\lambda_к/\lambda+\varepsilon\mu/\lambda)\,\theta}; \tag{9.81}$$

в устройстве с контролем дублированием

$$P_д(\theta) = e^{-\lambda\,(2+\lambda_к/\lambda)\,\theta}. \tag{9.82}$$

Время безотказной работы θ устройства определяется в зависимости от количества единиц оборудования устройства $Q_о$. Интенсивность сбоя μ и вероятности пропуска хотя бы одного сбоя $Q_о$ (θ) для устройства без контроля

$$\theta_о = Q_о(\theta)/(\lambda + \mu); \qquad (9.83)$$

для устройства с контролем по модулю

$$\theta_м \approx Q_о(\theta)/(\lambda + \gamma\lambda + \varepsilon\mu), \qquad (9.84)$$

где $\gamma = \lambda_к/\lambda$, $\lambda_к$ — интенсивность отказов аппаратуры контроля; для устройства при контроле дублированием

$$Q_д \approx Q_о(\theta)/(2\lambda + \gamma\lambda); \qquad (9.85)$$

для устройства при контроле троированием

$$\theta_т \approx Q_о(\theta)(1/3\lambda + 1/2\mu). \qquad (9.86)$$

Выигрыш в надежности работы устройства при различных видах контроля вычисляется по следующим формулам:
для контроля по модулю

$$\psi_м = \theta_м/\theta = \frac{1 + \mu/\lambda}{1 + \varepsilon\frac{\mu}{\lambda}\gamma}; \qquad (9.87)$$

для контроля дублированием

$$\psi_д = \theta_д/\theta = \frac{1 + \mu/\lambda}{2 + \gamma}; \qquad (9.88)$$

для контроля троированием

$$\psi_т = \theta_т/\theta = \frac{5 + 2\mu/\lambda + 3\lambda/\mu}{6}. \qquad (9.89)$$

Следует сказать несколько слов о реализации видов контроля.

Контроль по модулю 2 чаще всего осуществляют при пересылках информации и реализуют на специализированных микросхемах (например К155ИП2). В МПК БИС аппаратура контроля по модулю 2 заложена в специализированных передатчиках.

Контроль по модулю 3 дает наиболее приемлемые результаты по эффективности при минимальных затратах на оборудование. Схемы контроля по модулю 3 реализуют на каждом приемнике и передатчике. При этом виде контроля вместе с информационными разрядами передаются два контрольных (остаток от деления числа на 3). Их источник — передатчик информации. Приемник информации формирует два «своих» контрольных разряда и сравнивает их с принятыми.

Существуют методы получения контрольных разрядов при выполнении арифметических и логических операций.

Контроль дублированием осуществляют подключением двух одинаковых вычислительных устройств. Оба устройства выпол-

няют одну и ту же операцию, результаты сравниваются, и при совпадении происходит выдача результата. Этот метод используют при реализации не только контроля, но и для оперативного резервирования управляющего устройства. Различают «горячий» и «холодный» резервы. В первом случае происходит автоматическое переключение на резервную РЭА при обнаружении отказа и продолжение решения задачи на резервной РЭА. Во втором случае выполнение программы после обнаружения сбоя, как правило, осуществляют сначала.

Троирование относят к «интеллектуальным» методам контроля. Он позволяет исключить из системы контроля отказавшее звено.

В последние годы появились методы контроля, при которых можно выявить не только сбой в РЭА, но и исправить результат. Эти методы получили название методов самокорректирующихся кодов. Типичным является метод, использующий коды Хэминга. Он основывается на внесении значительной избыточности в РЭА за счет аппаратуры формирования специального кода, соответствующего определенным комбинациям разрядов.

Этот метод получил распространение в автономных средствах, работающих длительное время без вмешательства человека, например, в двигательных установках, устройствах управления энергетикой реакторов, автономных буях и др. Этот метод предложен сотрудником фирмы «Белл Леборейшенс» (США) Ричардом Хэмингом. Схема, реализующая этот метод, позволяет увеличить надежность плат памяти в 85 раз за период работы 10 000 ч. Этот метод позволяет обнаруживать и исправлять все однократные ошибки и обнаруживать двухкратные. Это позволяет микропроцессорной системе действовать эффективно в случаях возникновения неисправляемых однократных ошибок.

Рассмотрим на примере, как осуществляется контроль с помощью кодов Хэминга.

Сначала осуществляется кодирование (для байта данных) поступающих данных по табл. 9.2.

Пятиразрядная комбинация контрольных битов записывается совместно с байтом данных в ЗУ. При считывании аналогичным образом происходит формирование $C1'$ ... $C4''$, C'_x. Затем происходит суммирование по модулю 2 считанных из ЗУ контрольных разрядов и полученных. Сформированные таким образом пять сумм называются битами синдрома. Например, бит синдрома S получен суммированием по модулю 2 C_x и C'_x.

Согласно методу Хэминга, при $S_x = 1$ имеет место двухкратная ошибка. Если $S_x = 0$, то можно продолжить анализ на наличие ошибок по табл. 9.3.

После локализации ошибки «неисправные» разряды информации инвертируются и записываются в качестве верной информации на место неверной. Контрольные биты обычно не исправляются.

Таблица 9.2. Маска для генерации контрольных битов

Паритет	Контрольный разряд	Разряды данных							
		0	1	2	3	4	5	6	7
Нечетный	*C*1	×	×	×	—	×	—	×	—
Четный	*C*2	×	—	—	×	×	—	—	×
Нечетный	*C*3	—	×	—	—	—	×	×	×
Четный	*C*4	—	—	×	×	×	×	×	×
Нечетный	*CX*	×	×	×	×	—	×	—	—

Примечание. × — маскируемый разряд; бит *CX* — дополнение до нечетности остальных четырех контрольных битов *C*1 ... *C*4.

Таблица 9.3. Карта синдромов

S_4	S_3	S_2 S_1			
		0 0	0 1	1 0	1 1
0	0	Нет ошибок	*C*1	*C*2	*D*0
0	1	*C*3	*D*1	*	*
1	0	*C*4	*D*2	*D*3	*D*4
1	1	*D*5	*D*6	*D*7	*

Примечание. * — наличие многократной ошибки; *C*1 ... *C*4 — контрольные биты операнда; *D*0 ... *D*7 — биты операнда, где обнаружена ошибка.

Глава 10

СИСТЕМЫ ПОДДЕРЖКИ ПРОЕКТИРОВАНИЯ МИКРОПРОЦЕССОРНОЙ АППАРАТУРЫ

10.1. КЛАССИФИКАЦИЯ СИСТЕМ ПОДДЕРЖКИ

Использование МПК БИС, однокристальных и одноплатных микроЭВМ существенно меняет процесс разработки аппаратуры.

Процесс проектирования микропроцессорной аппаратуры включает: проектирование программного обеспечения, на которое направлены основные усилия разработчиков (его стоимость составляет до 90 % всей стоимости проектирования, при этом необходимы специальные средства его отладки); проектирование технических средств, которое осуществляют параллельно с проектированием программного обеспечения микропроцессорной аппаратуры, при этом трудоемкой и сложной является их совместная отладка.

Проектирование микропроцессорной аппаратуры состоит из следующих этапов [31]: формализация требований к аппаратуре; выбор микропроцессорных средств; разработка укрупненного алгоритма функционирования аппаратуры; определение функций, реализуемых аппаратно и программно; параллельная разработка технических средств и программного обеспечения микропроцессорной аппаратуры; комплексная отладка и совместные испытания технических средств и программного обеспечения.

Системы поддержки проектирования микропроцессорной аппаратуры (СППМА) предназначены для автоматизации разработки программного обеспечения, комплексной отладки и совместного испытания технических средств и программного обеспечения микропроцессорной аппаратуры.

СППМА можно разделить [57] на программные (кросс-системы), программно-аппаратные и аппаратно-программные системы поддержки.

Программные системы поддержки (или кросс-системы) представляют комплексы программ, написанных для универсальных ЭВМ, включающие кросс-ассемблеры, кросс-компиляторы и отладочные эмуляторы. Структуры и алгоритмы кросс-ассемблеров и кросс-компиляторов аналогичны структурам и алгоритмам ассемблеров и компиляторов, разработанных для универсальных ЭВМ. Отладочные эмуляторы содержат программы, моделирующие выполнение команд микропроцессорным устройством, и программы отладки, позволяющие следить за ходом моделирования испол-

нения разрабатываемой программы и вносить в нее изменения. Широкий набор внешних устройств и развитые операционные системы универсальных ЭВМ предоставляют разработчику кросс-систем и их пользователю большие возможности, а именно удобные методы создания и ведения библиотек отлаживаемых программ; стандартные средства ведения отладки; эффективное взаимодействие с ЭВМ; моделирование ситуаций, отладка которых на реальном объекте управления затруднена.

В настоящее время применяют специализированные программные средства, предназначенные для разработки программного обеспечения микропроцессорной аппаратуры стандартной архитектуры (например, одноплатные микроЭВМ со стандартной структурой технических средств и определенным интерфейсом сопряжения с внешними устройствами и управляющим объектом, а также типовым объемом и организацией памяти). В этом случае кросс-ассемблеры, кросс-компиляторы и отладочные эмуляторы настроены на фиксированную систему команд и предполагают наличие только одного программируемого компонента системы — микропроцессора. В универсальных кросс-системах (для микропрограммируемой микропроцессорной аппаратуры) допускается настройка ассемблеров, компиляторов и эмуляторов на любую систему команд. Это позволяет проводить отладку программного обеспечения микропроцессорных систем с различной архитектурой. Однако несмотря на то, что процесс настройки автоматизирован, он занимает от 3 человеко-месяцев до 3 человеко-лет, причем наиболее трудоемкой является настройка отладочного эмулятора.

К недостаткам кросс-систем относятся: высокая стоимость их использования; невозможность отладки технических средств и совместной отладки программного обеспечения и технических средств микропроцессорной аппаратуры.

В *программно-аппаратных* системах поддержки для получения программы в кодах микропроцессора также используют кросс-ассемблеры и компиляторы, но для исполнения этих программ применяют прототип микропроцессорной аппаратуры, работающей под управлением кросс-ЭВМ. Обладая достоинствами программных систем поддержки, программно-аппаратные системы позволяют отлаживать программы для микропроцессорной аппаратуры, работающей в реальном масштабе времени, а имеющиеся в таких системах внутрисхемные эмуляторы дают возможность отлаживать технические средства разрабатываемой аппаратуры. К недостатку программно-аппаратных средств разработки и отладки программного обеспечения следует отнести необходимость изготовления и предварительной отладки прототипа технических средств микропроцессорной аппаратуры. Кроме того, эти системы существенно сложнее, так как требуют специальной аппаратуры для организации совместной работы универсальной ЭВМ и микропроцессорной аппаратуры.

Аппаратно-программные системы поддержки проектирования (отладочные комплексы) на основе микроЭВМ обеспечивают параллельное проектирование и совместную отладку программного обеспечения и технических средств микропроцессорной аппаратуры. Возможности этих комплексов определяются объемом полупроводниковой памяти, наличием внешних ЗУ, набором периферийных устройств и составом системного программного обеспечения. Совершенные аппаратно-программные отладочные комплексы обеспечивают программирование как на ассемблере, так и на языках высокого уровня, оперативное взаимодействие с помощью дисплеев разработчика с микроЭВМ, хранение программ в ЗУ на магнитном диске. Стоимость таких комплексов определяется в основном стоимостью внешних устройств и, как правило, существенно меньше стоимости миниЭВМ. В эксплуатации эти комплексы дешевле универсальных ЭВМ.

В связи с этим при разработке микропроцессорной аппаратуры преимущественно используют отладочные комплексы.

10.2. СИСТЕМЫ ПОДДЕРЖКИ ПРОЕКТИРОВАНИЯ МИКРОПРОЦЕССОРНОЙ АППАРАТУРЫ НА ОСНОВЕ МИКРОПРОЦЕССОРОВ С ФИКСИРОВАННОЙ СИСТЕМОЙ КОМАНД

Системы поддержки охватывают микропроцессоры как с фиксированной системой команд (сюда входят однокристальные и одноплатные микроЭВМ), так и микропрограммируемые секционированные МП, позволяющие получать различные архитектуры. Системы поддержки для каждого класса имеют свои особенности и обычно ориентированы только на поддержку микропроцессора (или микропроцессоров) одного из классов. Поэтому рассмотрим системы поддержки отдельно для каждого класса МП.

10.2.1. Программные (кросс) системы поддержки

Эти системы позволяют проводить разработку и отлаживать программное обеспечение (ПО) микропроцессорной аппаратуры без микропроцессорных средств, используя специальные кросс-программы, называемые *системными*. (Если эти программы реализованы на той же микроЭВМ, для которой они вырабатывают программное обеспечение или входят в состав ПО аппаратно-программных комплексов, их называют *резидентными*). Рассмотрим основные из них [1].

Ассемблер. Машинно-ориентированный язык программирования, в котором используются мнемонические обозначения команд, отражающие их функции и символические имена переменных. Системная программа, транслирующая исходную программу, написанную на языке ассемблера, в коды команд микропроцессора (объектную программу), называется *ассемблером*. При трансляции в объектную программу каждая команда ассемблера преобразуется в двоичный код команды микропроцессора. Совершенный

ассемблер обычно содержит и макропроцессор-системную программу, позволяющую включать в разрабатываемую программу группу команд с помощью присвоенной ей мнемоники. Такой ассемблер называется *макроассемблером*.

Трансляторы с языков высокого уровня. Языки высокого уровня (ЯВУ) Бейсик, Фортран, Паскаль существенно упрощают написание программы. В качестве операторов используются специальные (ключевые) слова английского языка. Одному оператору на языке высокого уровня соответствует до нескольких десятков команд объектного кода, что сокращает трудоемкость программирования. Трансляторы с ЯВУ разделяют на компиляторы и интерпретаторы.

Компилятор — это системная программа, транслирующая исходную программу на ЯВУ в объектный код. При большом объеме программы ее текст на ЯВУ занимает в ЗУ меньше места, чем объектный код. При специальном кодировании операторов ЯВУ программу можно не транслировать в объектный код, а вводить построчно и одновременно выполнять. При этом отпадает необходимость хранения в памяти объектного кода всей программы. Системная программа, обеспечивающая выполнение наборов команд в соответствии с операторами исходной программы, называется *интерпретатором*. Наибольшее распространение получили интерпретаторы с языка Бейсик.

Ассемблеры и компиляторы могут генерировать код объектной программы в абсолютном и перемещаемом форматах. В первом случае в коды команд объектной программы заносят абсолютные адреса операндов и переходов, во втором — относительные по отношению к базовым адресам программных сегментов. Перемещаемая объектная программа может быть загружена для выполнения в любую область памяти, для этого программой-загрузчиком должна быть выполнена соответствующая коррекция относительных адресов.

Редактор текста. Системная программа, которая в соответствии с командами программиста, как правило, вводимыми с терминала, вносит исправления в текст исходной программы, находящейся в памяти ЭВМ.

Загрузчик и редактор связей. Загрузчик — это системная программа, с помощью которой объектная программа вводится в память ЭВМ. Если загружаемый объектный код является абсолютным, то он записывается по адресу, определенному при трансляции. Если объектный код является перемещаемым, загрузчик должен осуществить преобразование всех относительных адресных ссылок в абсолютные и загрузить полученную абсолютную объектную программу в требуемую область памяти.

При разработке больших программ обычно удобнее разбить программу на части (модули), которые транслируются отдельно. Полученные перемещаемые объектные модули затем объединяются в общую программу программой-редактором связей.

Отладочный монитор. Назначение отладочного монитора состоит в ускорении и облегчении поиска и исправления ошибок при отладке объектной программы. Монитор обеспечивает возможность останова выполнения отлаживаемой объектной программы в любой наперед заданной точке (точке разрыва). После останова программы монитор позволяет выводить на экран или на печать содержимое регистров и памяти и модифицировать его. В этом и заключается отладка.

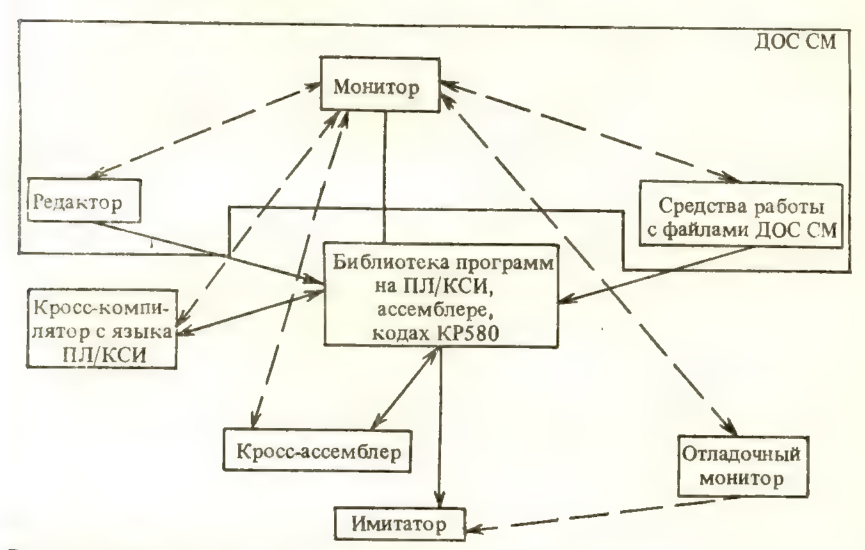


Рис. 10.1. Кросс-система разработки программного обеспечения микропроцессорных устройств КСИ-1

— — — — — управляющие связи; ——— — преобразование программ

Отладочные мониторы позволяют выполнять программу также и в пошаговом режиме, т. е. с остановом после выполнения каждой команды. Отладочный монитор должен находиться в памяти ЭВМ вместе с отлаживаемой программой.

Эмулятор. Системная программа, моделирующая выполнение команд программы микропроцессорной аппаратуры и позволяющая проводить отладку этих программ без микропроцессорной аппаратуры.

Перед написанием программ для микропроцессорной аппаратуры разработчик должен выбрать язык программирования. В качестве примера рассмотрим кросс-систему КСИ-1 поддержки проектирования микропроцессорной аппаратуры на основе МПК БИС серии КР580, реализованной на мини-ЭВМ СМ-4 [32].

Система КСИ-1 (рис. 10.1) функционирует в рамках дисковой операционной системы *ДОС СМ* и фактически является ее подсистемой, поэтому КСИ-1 использует редактор текста и средства

ведения библиотек *ДОС СМ*. Все программы системы КСИ-1 написаны на Фортране.

Язык ПЛ/КСИ, используемый в КСИ-1, входит в подмножество языка программирования ПЛ/М-80 и имеет следующие ограничения:

отлаживаемая программа представляет собой один модуль, все переменные являются глобальными;

в языке используются простые переменные и одномерные массивы. Переменные и элементы массивов являются целыми числами без знака и могут иметь тип *BYTE* (занимают 1 байт) или *ADDRESS* (занимают 2 байта);

в операторе присваивания допускается только одно действие;

идентификаторы содержат не более пяти символов.

В языке используются следующие операторы: присваивания, *DECL* (Декларация), *IN* (Ввод), *OUT* (Вывод), простые и итеративные операторы *DO* (Выполнить), операторы *END* (Конец), *IF* (Если), *GOTO* (Перейти к), *ORG* (Начальный адрес). В операторах *IN* и *OUT* указываются идентификатор передаваемой простой переменной или элемента массива (при выводе возможно указание константы) и номер устройства ввода или вывода. Оператор *IF* имеет вид: *IF* (переменная) *THEN* (оператор). В качестве переменной может использоваться простая переменная или элемент массива. Операторами могут быть *IN*, *OUT*, *DO*, *IF*, *GOTO*, операторы присваивания. Если переменная имеет в младшем бите единицу, то оператор, следующий после слова *THEN*, выполняется. Оператор *ORG* служит для задания адреса размещения программы.

Язык программирования ПЛ/КСИ может использоваться совместно с ассемблером. Так, при разработке программы возможно написание одних ее частей на ассемблере, а других на ПЛ/КСИ. Компилятор с языка ПЛ/КСИ транслирует фрагменты, написанные на этом языке (каждый фрагмент должен являться простым блоком *DO*), в ассемблер микропроцессора КР580ИК80А и совмещает полученные фрагменты с частями программы, написанными непосредственно на ассемблере.

Компилятор с языка ассемблера КСИ-1 генерирует абсолютный объектный код.

Ассемблер не позволяет использовать макроопределения и условное ассемблирование, в остальном он соответствует языку ассемблера микропроцессора 8080 фирмы «Интел». Транслятор с языка ассемблера занимает 20К слов.

Объектные коды отлаживаемой программы, полученные в результате трансляции кросс-ассемблером, размещаются в младших байтах двух байтовых слов массива *MEM*. Старшие байты содержат семь однобитовых признаков: отладочный вывод содержимого регистров или заданных ячеек памяти после выполнения данной команды; прерывание программы после выполнения данной команды; байт данных; первый байт команды; метки; кон-

трольный останов после выполнения данной команды; защита памяти на запись.

Старший бит 16-разрядного слова всегда имеет значение 0. Значение индекса массива *MEM* является адресом ячейки памяти. Кроме того, в базу данных имитатора входят: переменные, моделирующие счетчик команд PC; указатель стека *SP*; накопитель *A*; шесть регистров общего назначения *B*, *C*, *D*, *E*, *H*, *L*; однобитовые признаки результата *S*, *Z*, *P*, *C*, *AC*, триггер разрешения прерывания *TI*; счетчик тактов *T*. При выполнении каждой команды отлаживаемой программы к содержимому счетчика тактов прибавляется время выполнения команды микропроцессором в машинных тактах. При прерывании, контрольном останове или после выполнения программы содержимое счетчика тактов выводится на экран дисплея. В базу данных входят также стек для запоминания состояния программы; таблица адресов переменных, выводимых при трассировке; таблица меток, создаваемая кросс-ассемблером, и признаки отладочных режимов: *PPM* — признак выдачи содержимого ячеек ЗУ; *PPR* — признак выдачи содержимого регистров; *STEP* — признак выполнения одной команды; *STATE* — признак запоминания состояния; *TRAS* — номер режима трассировки.

Интерпретатор представляет собой набор подпрограмм, соответствующих группам команд микропроцессора. Эти подпрограммы объединяются в одно целое программой дешифрации команд. Интерпретатор начинает свою работу при задании пользователем директивы *E* (Выполнить) или *S* (Шаг). Аргументом директивы *E* является адрес запуска программы. Адрес заносится в ячейку памяти, моделирующую счетчик команд микропроцессора. Код команды, расположенный по введенному адресу, выбирается из массива *MEM* и дешифрируется. Далее из массива *MEM*, если это необходимо, выбираются остальные байты команды, и она выполняется. При выполнении команды анализируются заданные режимы трассировки и состояния различных признаков (например, признака контрольного останова). Если это необходимо, запоминается состояние программы, выводится содержимое регистров и ячеек памяти и т. д.

При написании имитатора использовалась система перекрытий ДОС СМ. Подпрограммы имитации системы команд разбиты на 20 файлов, размещаемых на магнитном диске.

Резидентная часть имитатора составляет 15 К слов ОЗУ, каждое перекрытие — до 5 К слов. Общий объем имитатора около 115 К слов. Отладочный монитор обеспечивает взаимодействие пользователя с кросс-системой путем интерпретации директив, вводимых с клавиатуры дисплея. Отладочный монитор и имитатор используют одну и ту же базу данных.

Все директивы монитора делят на управляющие и отладочные. Управляющих директив две: *R* (Чтение ассемблированного модуля в массив *MEM*) и *O* (Переход в отладочный режим). По

директиве *R* объектная программа, полученная с помощью транслятора с языка ассемблер или введения в библиотеку программ в двоичном коде, переписывается в ОЗУ СМ-4. Директива *О* осуществляет передачу управления интерпретатору отладочных директив.

Описанная кросс-система позволяет разрабатывать и отлаживать программное обеспечение микропроцессорных устройств, построенных на основе МПК БИС КР580. Компиляторы с языков ассемблер и ПЛ/КСИ обеспечивают высокую производительность труда программистов. Различные режимы трассировки, средства анализа и модификации содержимого регистров и ячеек памяти, возможность запоминания и восстановления состояния программы позволяют эффективно проводить отладку в оперативном режиме.

Однако эффективность процесса разработки программного обеспечения с использованием КСИ-1 можно повышать следующими ее усовершенствованиями:

реализацией компилятора с языка ПЛ/М без каких-либо ограничений, созданием макроассемблера, генерирующего перемещаемый код;

введением в систему монитора, обеспечивающего удобное задание исходных данных и ожидаемых результатов, тестовых примеров;

настройкой эмулятора на конкретную структуру технических средств с учетом программируемых периферийных БИС (БИС параллельных и последовательных интерфейсов, контроллеров прямого доступа в память, БИС подсистемы прерываний, таймеров);

введением возможности эмуляции мультимикропроцессорных систем.

10.2.2. Аппаратно-программные системы поддержки

Все аппаратно-программные комплексы для отладки микропроцессорных систем можно разделить на четыре класса [31]: оценочные модули; аппаратно-программные отладочные комплексы ограниченной мощности; развитые аппаратно-программные отладочные комплексы; универсальные аппаратно-программные отладочные комплексы.

Оценочные модули. Системы проектирования этого класса представляют собой одноплатные микроЭВМ с минимальной конфигурацией аппаратного обеспечения, к которым в качестве устройств ввода-вывода подключается простейшая клавишная панель с индикаторами, а в ряде случаев телетайп. Основой оценочных модулей могут служить как одноплатные микроЭВМ, предназначенные для встраивания в различное оборудование, так и микроЭВМ, специально спроектированные для этой цели. Оценочные модули предназначены для оценки возможностей

микропроцессорных средств, а также для разработки прототипов несложных микропроцессорных устройств с объемом программного обеспечения 1—2 Кбайт.

Емкость ПЗУ для хранения системных программ составляет от 256 байт до 4 Кбайт; емкость ОЗУ, в котором хранятся данные и отлаживаемые программы пользователя, от 128 байт до 4 Кбайт. Как правило, на плате предусмотрен также интерфейс последовательного асинхронного канала ввода-вывода для подключения телетайпа и до 48 линий параллельного ввода-вывода для сигналов управляемого объекта. Как и в обычных одноплатных микроЭВМ, имеется возможность устанавливать в свободные гнезда на плате дополнительные ИС памяти и интерфейса ввода-вывода, а в ряде случаев подключать дополнительные платы — ЗУ и интерфейса. Шины данных и адресные шины микропроцессора, сигналы управления и синхронизации, а также интерфейс ввода-вывода выводятся на разъем печатной платы, что дает возможность разработчику подключать дополнительную память и производить сопряжение с терминалами.

Серийные одноплатные микроЭВМ используют как средства проектирования совместно с дополнительными платами, на которых расположены клавиши управления и шестнадцатиричная клавиатура для ввода и модификации программы пользователя, а также светоизлучающие диоды (СИД) и сегментные индикаторы для наблюдения за внутренними состояниями процессора, сигналами шин и данными в памяти. С основной платой микроЭВМ такие платы соединяются многопроводным гибким кабелем. В случае одноплатных микроЭВМ, предназначенных специально для разработки программ, клавиатура и устройства индикации расположены на той же плате.

При использовании систем проектирования данного класса программирование осуществляют в машинных командах, что эффективно при объеме программ до 256 байт. Применяя технику программирования в машинных командах, можно вносить изменения и в более сложные программы.

Программное обеспечение оценочных наборов, как правило, ограничивается программой-монитором, размещенной в ПЗУ. Однако, несмотря на небольшой объем (256 байт — 2 Кбайт), такой монитор предоставляет достаточно гибкие средства для отладки прикладных программ, а именно: пошаговый режим, задание адресов точек разрыва, вывод на светоизлучающие диоды и изменение с клавиатуры содержимого регистров и ячеек ОЗУ, работу с командными клавишами и телетайпом.

В качестве примера рассмотрим оценочный модуль фирмы «ХЕК Электроник» (ФРГ), являющийся типичным аппаратно-программным комплексом этого класса. Конструктивно модуль оформлен в виде печатной платы, на которой смонтированы комплект микропроцессорных БИС, дополнительные схемотехнические элементы, шестнадцатиричная клавиатура и восемь се-

мисегментных индикаторов. Плата имеет разъем, через который разработчик может самостоятельно подключать дополнительную память и терминальные устройства.

Центральным процессором служит 8-разрядный прибор 8080 фирмы «Интел».

При включении питания генерируется сигнал «Сброс». Этот сигнал устанавливает счетчик команд процессора в нулевое состояние, что означает безусловное выполнение программы, начиная с ячейки с нулевым адресом. Пошаговый режим выполнения программы, облегчающий ее отладку, реализуется с помощью сигналов «Готов» и «Ждать». При отсутствии сигнала «Готов» процессор находится в состоянии ожидания, о чем свидетельствует сигнал «Ждать» на выходе. При нажатии кнопки «Шаг» на вход «Готов» поступает короткий импульс, разрешающий выполнение следующего шага программы.

Запоминающее устройство состоит из ОЗУ емкостью 512 байт с возможностью расширения до 1 Кбайт (на плате предусмотрены гнезда для установки дополнительных ИС памяти) и ЭППЗУ емкостью 1 Кбайт для хранения программы-монитора. ОЗУ постоянно подключено к отдельному аккумуляторному источнику питания +5 В. Это позволяет сохранять информацию в ЗУ в течение нескольких недель.

Модуль имеет клавиатуру из 9 командных клавиш и 16 клавиш для ввода шестнадцатиричных чисел. При нажатии на одну из командных клавиш вызывается соответствующая подпрограмма монитора. Результаты работы подпрограммы отображаются на восьми семисегментных индикаторах, образующих индикаторный регистр. Старшие четыре разряда индикаторного регистра отображают адрес, младшие два разряда — содержимое ячейки памяти.

В качестве оценочного модуля можно рассматривать и автономное отладочно-учебное устройство «Электроника ОУ-580», предназначенное для обучения и отладки микроконтроллеров на БИС серии КР580.

Недостатками оценочных модулей как средств проектирования являются ограниченные возможности ввода-вывода, необходимость программирования в машинных кодах и невозможность разработки программ большого объема.

Аппаратно-программные комплексы ограниченной мощности. Отличаются от оценочных модулей программным обеспечением, объемом памяти и усложненным интерфейсом, позволяющим использовать более широкий диапазон ввода-вывода. Поставляются в виде конструктивно законченного блока, объединяющего 2 ... 15 плат и имеющего пульт с установленными на нем клавишами управления и индикацией. Так же как в оценочном модуле ядро системы составляет микроЭВМ на основе комплекта БИС, который будет применяться в разрабатываемой аппаратуре.

Запоминающее устройство в большинстве случаев состоит из ПЗУ или ППЗУ емкостью от 1 до 4 Кбайт с возможностью расширения до 64 Кбайт.

Увеличение объема ЗУ по сравнению с оценочными модулями обусловлено бо́льшими объемами системного программного обеспечения и отлаживаемых программ.

Системное программное обеспечение кроме отладочного монитора включает загрузчик, редактор текста, программы обслуживания периферийных устройств, ассемблер. В ряде случаев в ПЗУ записывают комплекс программ, представляющий собой несложную операционную систему. Такой состав системного программного обеспечения позволяет пользователю эффективно отлаживать программы объемом 3—4 Кбайт или более. Программирование ведется на ассемблере. Программа-ассемблер в простейших системах данного класса обеспечивает получение абсолютного объектного кода и в более сложных — перемещаемого кода.

Самым дешевым и доступным устройством ввода-вывода, совмещающим операции с перфолентой и вывод на печать в буквенно-цифровом виде, является телетайп. В связи с ограниченным объемом ЗУ одновременное хранение всего системного программного обеспечения и разрабатываемой программы пользователя на ассемблере в ЗУ невозможна. Поэтому обычно в ПЗУ находятся только программа-монитор и программа-загрузчик. Ассемблер и редактор текста расположены на перфоленте и при необходимости поочередно загружаются в ОЗУ с помощью телетайпа.

Отлаживаемая программа в символическом виде вводится и обрабатывается ассемблером построчно. Ассемблер обычно является двухпроходным, а для распечатки программы может использоваться третий проход. В связи с этим каждую программу на ассемблере необходимо вводить с перфоленты 2—3 раза.

Низкая скорость работы телетайпа делает эту процедуру весьма длительной. Так, для получения объектного кода программы длиной 100 команд с помощью трехпроходного ассемблера требуется не менее 20 мин. Распечатка двух страниц листинга занимает более 8 мин. В более совершенных отладочных комплексах предусмотрено высокоскоростное перфоленточное устройство ввода-вывода и отдельное печатающее устройство. Производительность отладки при этом может возрасти в 16 раз, что не исключает многократного ввода системных и прикладных программ с перфолент.

Существенным шагом вперед по сравнению с телетайпом является также подключение к отладочному комплексу ленточного кассетного ЗУ. В целях уменьшения стоимости такое ЗУ часто строится на основе обычного звукозаписывающего магнитофона. При этом ассемблер и редактор текста, а иногда и отлаживаемая программа размещаются в этом ЗУ, что существенно повышает удобство пользования отладочным комплексом.

Для увеличения производительности труда разработчика к аппаратно-программному комплексу в ряде случаев подключают буквенно-цифровой дисплей с клавиатурой, что удобно для оперативного отлаживания программ.

Более подробно возможности отладочных комплексов ограниченной мощности рассмотрим на примере системы проектирования *CDP*18S004 фирмы «РСА», предназначенной для отладки программного обеспечения микропроцессора серии 1800.

Технические средства отладочного комплекса размещены на 11 печатных платах, содержащих процессор 1802, генератор синхроимпульсов, схемы управления, ОЗУ на 4 Кбайт, шинные формирователи, дешифратор ввода-вывода, порт ввода-вывода, интерфейсы телетайпа, кассетное ленточное ЗУ и ПЗУ емкостью 1,5 Кбайт с отладочным монитором. Печатные платы вместе с источником питания помещены в корпус, предусмотрены места для дополнительных плат ЗУ и портов ввода-вывода. На передней панели установлены следующие клавиши управления: Включение питания, Сброс, Пуск программы, Пуск монитора, Загрузка. Системное программное обеспечение состоит из редактора текста, ассемблера и отладочного монитора.

Для управления работой редактора текста предусмотрены следующие команды: УДАЛИТЬ, ДОБАВИТЬ, ВСТАВИТЬ, ЗАПОМНИТЬ, НАЙТИ И ЗАПОМНИТЬ, ОТПЕЧАТАТЬ, ВЫДАТЬ. Отладочный монитор позволяет по командам пользователя проверять и модифицировать содержимое ячеек ЗУ и регистров процессора, устанавливать и удалять точки разрыва программы пользователя, выполнять заданное число команд, запускать и продолжать выполнение отлаживаемой программы.

Редактор текста и ассемблер могут размещаться на перфоленте или магнитной ленте. Эти две программы вводятся в ОЗУ поочередно. При минимальной емкости ОЗУ, кроме редактора текста имеется рабочий буфер для программы пользователя емкостью 1 Кбайт. Ассемблер позволяет транслировать в машинные коды прикладные программы, имеющие около 100 меток.

Развитые аппаратно-программные отладочные комплексы. Их принципиальными отличиями являются: внутрисхемная эмуляция отлаживаемых программ; возможность программирования на языках высокого уровня; широкий набор внешних устройств, включая накопители на гибких магнитных дисках (НГМД); наличие развитой дисковой операционной системы (ДОС).

Разработка микропроцессорных устройств может быть проведена с наименьшими затратами в минимальные сроки, если на протяжении всего цикла проектирования существует взаимосвязь между проектированием аппаратных и программных средств.

Если отладку программного обеспечения проектируемой микропроцессорной системы проводить отдельно от разработки ее технических средств, то совместная проверка технических средств системы и отлаженных программ может выявить их неправиль-

ную работу. Это связано с тем, что отладочные аппаратно-программные комплексы построены на основе стандартных микро-ЭВМ, не отражающих специфики проектируемого устройства. При этом трудно определить, чем вызвана неправильная работа устройства: неисправностью технических средств или невыявленными ошибками в программе.

При совместной отладке аппаратного и программного обеспечения к прототипу технических средств системы на место процессора с помощью гибкого кабеля подключают отладочный комплекс. При отладке программного обеспечения в этом случае используют процессорную плату и ЗУ отладочного комплекса, правильная работа которых гарантируется. Эта часть микропроцессорной системы функционирует вместе с разрабатываемым прототипом, что позволяет проводить отладку в реальных условиях. Прототип микропроцессорной системы может быть подключен к объекту, для управления которым система предназначена. Проверку программного обеспечения производят посредством отладочного монитора, хранящегося в ЗУ отладочного комплекса.

Такая замена процессора прототипа специальным отладочным устройством, которое может строиться как на основе микропроцессора, так и на обычных логических ИС, называется *внутрисхемной эмуляцией.*

При отладке необходимо иметь возможность после выполнения любой команды выводить на индикацию и модифицировать содержимое внутренних регистров и ячеек памяти, восстанавливать состояние процессора, имевшее место до выполнения некоторого количества последних команд, производить останов по заданному адресу и т. д. Все эти операции могут быть выполнены программно с помощью отладочного монитора, что обычно и делается в отладочных комплексах ограниченной мощности. В этом случае после выполнения каждой команды отлаживаемой программы осуществляется ее прерывание и переход к программе монитору. Последняя обеспечивает сравнение текущего адреса с заданными адресами остановов, вычисляет и запоминает содержимое внутренних регистров, не имеющих внешних выводов, осуществляет расчет времени, затрачиваемого на выполнение программы. Операции, выполняемые программой-монитором в этом случае, занимают существенно больше времени, чем работа отлаживаемой программы, вследствии чего отладка в реальном масштабе времени невозможна.

Внутрисхемные эмуляторы (ВСЭ) представляют собой устройства, в которых операции, необходимые для отладки, осуществляются аппаратно, что может быть выполнено без дополнительных затрат времени.

Для облегчения отладки технических средств системы аппаратно-программные отладочные комплексы рассматриваемого класса обычно предусматривают подключение логических анали-

заторов. Эти приборы запоминают и выводят на экран электронно-лучевой трубки (ЭЛТ) логические состояния выбранных шин для нескольких десятков тактов.

Развитые аппаратно-программные отладочные комплексы обычно предназначены для разработки систем на базе не одного микропроцессора, а семейства схожих микропроцессорных комплектов, выпускаемых одной фирмой. Разработку прототипа микропроцессорной системы с использованием таких отладочных комплексов осуществляют следующим образом:

разрабатываются схемы сопряжения с управляемым объектом и вспомогательные логические схемы обслуживания процессора, которые не являются типовыми;

к разработанной части опытного образца посредством гибкого кабеля подключают внутрисхемный эмулятор, при этом используют память и стандартные устройства ввода-вывода отладочного комплекса;

опытный образец проверяют с помощью диагностических средств отладочного комплекса и отлаживают совместную работу программного обеспечения и технических средств;

в опытный образец в определенной последовательности вводят остальные части технических средств — память, устройства ввода-вывода. По мере ввода отсоединяют соответствующие ресурсы отладочного комплекса. При этом комплексно отлаживают программные и технические средства.

При запуске опытного образца в производстве такой отладочный комплекс можно использовать для выполнения производственных испытаний.

Высокая эффективность развитых аппаратно-программных отладочных комплексов во многом обусловлена наличием разнообразных внешних устройств, в частности, дисплеев и НГМД. НГМД существенно увеличивает емкость ЗУ, что позволяет, во-первых, использовать трансляторы с языков высокого уровня — ПЛ/1, Фортран, Кобол и других, во-вторых, создать библиотеку отлаживаемых программ и стандартных подпрограмм. Трансляторы с языка ассемблера также широко используются в отладочных комплексах этого класса, однако они, как правило, генерируют перемещаемый код, что позволяет отлаживать программы большого объема по частям. Рассмотрим конкретный пример такого комплекса.

Развитый аппаратно-программный отладочный комплекс Интеллек МДС-800 предназначен для разработки и отладки как программного обеспечения, так и технических средств микропроцессорных систем на основе МП 8080, 8085, 3000 фирмы «Интел». Он может быть использован также для разработки систем на основе отечественных комплексов БИС серий КР580 и К589.

Комплекс представляет собой специализированную микроЭВМ на базе МП 8080 фирмы «Интел» (рис. 10.2) и включает *модуль центрального процессора*, *модуль ОЗУ* емкостью 16 Кбайт, *модуль*

монитора и *панель управления.* Для расширения возможностей комплекса к нему могут быть добавлены *модули ППЗУ*, до трех модулей ОЗУ емкостью по 16 Кбайт, *НГМД*, *модули внутрисхемных эмуляторов* ICE-80 и ICE-30. В состав аппаратного обеспечения комплекса входит набор стандартных внешних устройств: *телетайп*, *дисплей*, *фотосчитыватель*, *ленточный перфоратор*, *печатающее устройство*, *программатор ПЗУ*.

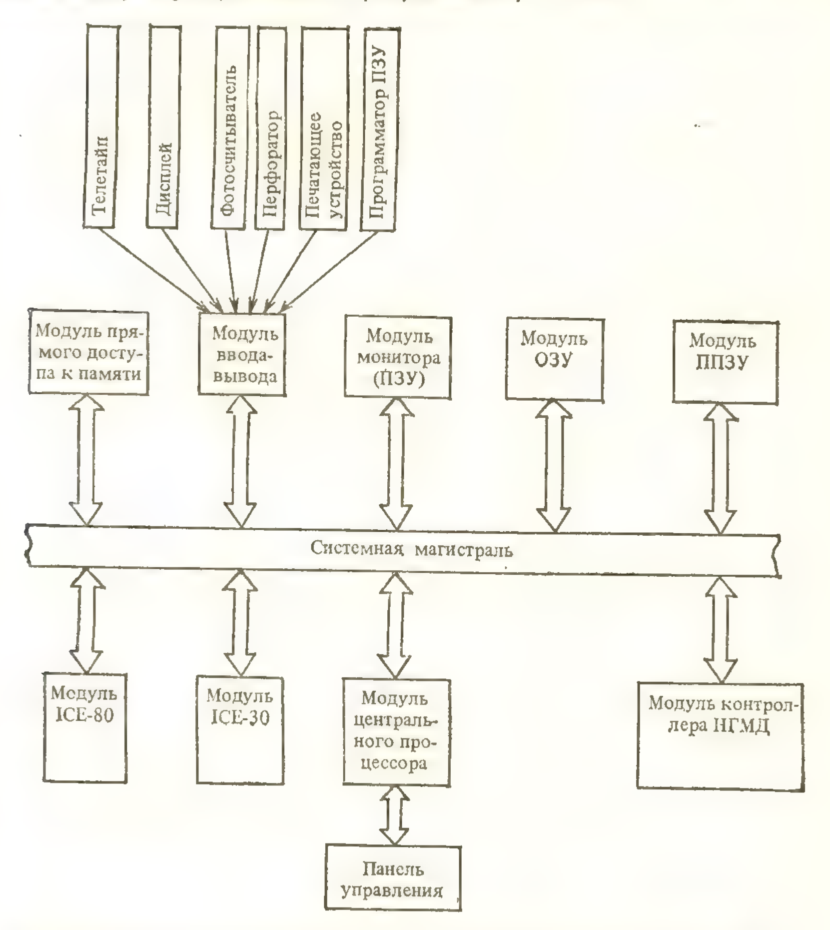


Рис. 10.2. Структурная схема отладочного комплекса Интеллек МДС-800

Основным устройством, с помощью которого осуществляется взаимодействие пользователя с комплексом, является пультовой терминал, в качестве которого может использоваться алфавитно-цифровой дисплей или телетайп.

Как правило, для отладки микропроцессорных систем применяют полный комплект Интеллек МДС-800 с объемом ОЗУ

64 Кбайт, НГМД, модулем внутрисхемного эмулятора. В качестве пультового терминала используют алфавитно-цифровой дисплей.

Комплекс МДС-800 включает средства для разработки как программного обеспечения, так и технических средств микропроцессорных систем. Для разработки и отладки программного обеспечения служат программа-монитор и программы ДОС. Для комплексной отладки программных и технических средств используют модуль внутрисхемной эмуляции и соответствующее программное обеспечение, также входящее в состав операционной системы. Программное обеспечение комплекса совместно со средствами внутрисхемной эмуляции позволяет осуществлять: запись программ (текстовых файлов) на том или ином входном языке, их редактирование, компиляцию; хранение исходных и объектных программ; выполнение и отладку программ, а также отладку технических средств.

Программа-монитор постоянно расположена в ПЗУ и позволяет выполнять ряд функций, обеспечивающих взаимодействие с пультовым терминалом, загрузку и исполнение объектных программ, а также их контроль и отладку.

Основу системного программного обеспечения комплекса составляет ДОС *ISIS-II*, которая позволяет работать с большими объемами информации, записываемой на НГМД, и представляет собой набор системных программ, необходимых для разработки программного обеспечения и помогающих разработчику управлять ресурсами комплекса при разработке, отладке и выполнении целевых программ.

В МДС-800 применяется принцип модульного программирования. Модульное программирование предполагает написание программы в виде достаточно простых модулей (файлов), которые впоследствии используются основной программой пользователя. Мнемонические имена файлов определяет пользователь, а ДОС обеспечивает работу пользователя с файлами на мнемоническом уровне.

Основным инструментом для создания и редактирования тестовых файлов в коде КОИ-7 является программа-редактор текста.

Процедура редактирования заключается в запуске программы-редактора, введении текста, его редактировании (замене, удалении или добавлении некоторых символов) и выводе текстового файла на печать или записи его на диск.

В состав ДОС входит широкий набор команд управления файлами. Эти команды позволяют копировать, удалять файлы, преобразовывать коды, просматривать оглавление диска. Программные модули могут быть написаны на различных языках. Каждый программный модуль может быть записан, откорректирован и отлажен независимо от других. Преобразование исходных текстовых файлов в объектные программы осуществляется компиляторами, входящими в состав ДОС. В операционную систему

включены программа-макроассемблер, компиляторы с языков высокого уровня (ПЛ/М-80, Паскаль, Фортран, язык СИ). Кроме того, имеется интерпретатор программ на языке Бейсик.

Полученные в результате компиляции файлы объектного кода имеют перемещаемый формат. Для компоновки программ из перемещаемых объектных модулей и размещения программ в ОЗУ в операционной системе имеются три системные программы: *LINK* (Редактор связей), *LOCATE* (Загрузчик) и *LIB* (Библиотекарь).

Программа-библиотекарь обеспечивает введение и хранение файлов объектных модулей. Эти модули могут быть использованы для построения более сложных программ. Для этого требуемые модули должны быть объединены в общую программу редактором связей, который объединяет объектные модули путем согласования относительных адресов межмодульных ссылок. Программа-загрузчик позволяет разместить полученную в перемещаемом формате объектную программу с учетом реального распределения адресов ОЗУ и ПЗУ, имеющихся в разрабатываемой микропроцессорной системе, что осуществляется введением соответствующих управляющих параметров в программу.

Программа-загрузчик присваивает абсолютные адреса перемещаемым объектным файлам, после чего полученные абсолютные объектные программы готовы для исполнения. Исполнение любой программы можно инициализировать, указав имя соответствующего файла.

Директива DEBUG (Отладка) дает возможность пользователю осуществлять загрузку и исполнять программы под управлением монитора.

Разработка программного обеспечения и технических средств при проектировании микропроцессорной системы должна осуществляться пользователем (группой пользователей) параллельно. Результатом этой работы является прикладная объектная программа, логическое функционирование которой проверено на системе МДС-800, а также микропроцессорная система-прототип. Система прототип представляет собой микропроцессорную систему на ранних стадиях разработки, в которую из всех возможных блоков должны входить, по крайней мере, центральный процессорный элемент и схемы сопряжения с шинами управления, адресов и данных. Дальнейшая отладка как программного обеспечения, так и технических средств на реальном макете может быть проведена с помощью внутрисхемного эмулятора *ICE*-80 без какого-либо дополнительного отладочного оборудования.

Внутрисхемный эмулятор *ICE*-80 включается вместо МП системы-прототипа и выполняет все функции — осуществляет эмуляцию. В то же время специальные программно-аппаратные средства, заложенные в блок внутрисхемной эмуляции, дают возможность пользователю активно контролировать ход вычислительного процесса в системе-прототипе. Внутрисхемный эмуля-

тор позволяет осуществлять исполнение, контроль и отладку программ в системе-прототипе в диалоговом режиме с терминала отладочного комплекса. При этом возможна отладка в реальном масштабе времени и в реальных условиях работы микропроцессорной системы.

Одним из важнейших качеств внутрисхемного эмулятора является возможность эмуляции им не только функций центрального процессорного элемента, но и функций ОЗУ (ППЗУ, ПЗУ) и периферийных устройств системы-прототипа. Для того чтобы некоторая часть или вся память системы прототипа, а также часть или все внешние устройства системы-прототипа были эмулированы памятью и внешними устройствами комплекса, внутрисхемным эмулятором используется механизм виртуальной адресации. Сущность этого механизма заключается в том, что перед началом эмуляции программой *ICE*-80 пользователю дается возможность установить соответствие между виртуальными (логическими) адресами блоков памяти и внешних устройств, используемых в его программах, и их физическими адресами в системе-прототипе или отладочном комплексе.

Внутрисхемный эмулятор позволяет предоставить в распоряжение разрабатываемой системы те аппаратные ресурсы, которые на этапе разработки в ней еще отсутствуют, но которыми обладает отладочный комплекс (память, внешние устройства). При этом система-прототип может быть подключена к управляемому объекту.

Директивы внутрисхемного эмулятора позволяют производить в реальном масштабе времени при подключенном к системе-прототипу управляемом объекте все отладочные операции, обычно выполняемые монитором: инициировать исполнение программы с указанием точек разрыва (останова), просматривать и редактировать содержимое ячеек ЗУ и регистров МП и т. п.

Таким образом, аппаратно-программный отладочный комплекс Интеллек МДС-800 позволяет одновременно и на реальном макете осуществлять отладку как программного обеспечения, так и технических средств микропроцессорной системы.

Внутрисхемный эмулятор выполнен на базе МП 8080 и является по отношению к центральному процессору отладочного комплекса самостоятельным внешним устройством. Обмен информацией между процессором комплекса и процессором эмулятора осуществляется с помощью специально выделенной области ОЗУ комплекса, называемой памятью обмена. Через память обмена ДОС *ISIS*-II посылает внутрисхемному эмулятору директивы и начальные параметры, инициирующие его работу, анализирует его состояние, принимая информацию о результатах выполнения каждой директивы. В свою очередь блок внутрисхемной эмуляции принимает от ДОС директивы пользователя и исполняет их, формируя при этом все информационные и управляющие сигналы для системы-прототипа.

Внутрисхемный эмулятор может работать в режимах опроса, эмуляции и пошаговом режиме.

В режиме опроса процессор комплекса производит ввод директив и начальных параметров в память обмена, анализ результатов эмуляции, осуществляет также пересылку данных между системой-прототипом и отладочным комплексом.

В режиме эмуляции выполнение программы происходит в системе-прототипе. При этом внутрисхемный эмулятор посылает в систему-прототип все сигналы, которые посылал бы в нее реальный МП 8080. Процесс эмуляции осуществляется до тех пор,

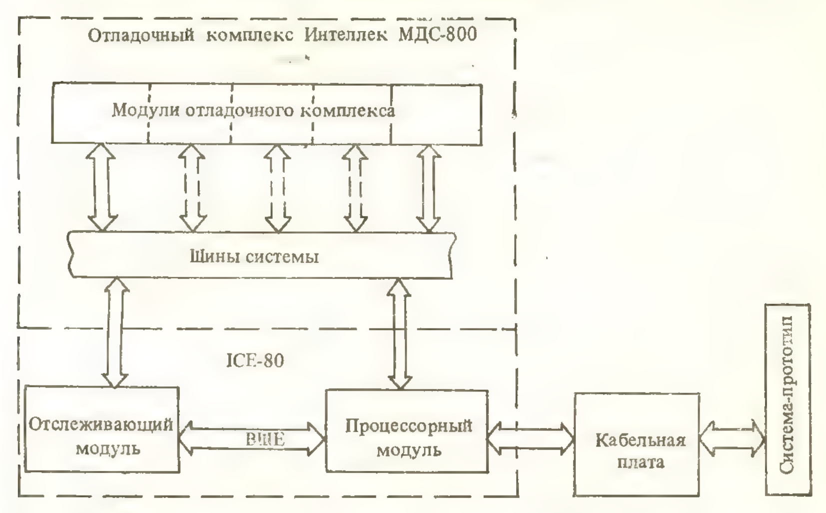


Рис. 10.3. Организация внутрисхемного эмулятора ICE-80 и его положение в системе

пока он не будет аппаратным способом остановлен эмулятором и программа *ICE*-80, расположенная в ОЗУ комплекса и сканирующая память обмена, не обнаружит, что достигнуты точки останова, указанные пользователем. После этого эмулятор переходит в режим опроса.

В пошаговом режиме после выполнения каждой команды процессор останавливается и на экран дисплея выдается содержимое регистров процессора, адресов ЗУ, к которым производилось обращение. Следующая команда выполняется по инструкции пользователя. При пошаговом режиме с автопродвижением внутрисхемный эмулятор автоматически переходит от команды к команде с задержкой 0,25 с.

Функционально внутрисхемный эмулятор *ICE*-80 (рис. 10.3) состоит из *процессорного модуля*, *отслеживающего модуля*, *кабельной платы* и разъема для подключения эмулятора к *системе-прототипу*. *Процессорный* и *отслеживающий модули* эмулятора выполнены в виде стандартных печатных плат, расположенных

в корпусе *отладочного комплекса*. К системе-прототипу внутрисхемный эмулятор подключается с помощью гибкого плоского кабеля с разъемом. *Кабельная плата* расположена непосредственно на кабеле, соединяющем платы эмулятора с *системой-прототипом*. Взаимодействие между платами внутрисхемного эмулятора осуществляется по внутренней шине эмулятора *ВШЭ*.

Системные функции между процессорным и отслеживающим модулями внутрисхемного эмулятора распределены следующим образом. Отслеживающий модуль обеспечивает прием команд от комплекса и инициирует исполнение процессорным модулем соответствующих подпрограмм обслуживания, записанных в ПЗУ отслеживающего модуля. Отслеживающий модуль обеспечивает также запоминание информации о выполнении 44 последних машинных циклов эмуляции.

В функции отслеживающего модуля входит анализ кодов адресов, команд, данных, кодов управляющего слова, формируемых во время эмуляции процессором системы-прототипа, а также останов эмуляции в случае обнаружения условий, указанных пользователем. Процессорный модуль во время эмуляции предоставляет свой МП системе-прототипу, обеспечивая возможность выполнения программ пользователя. Во время останова эмуляции процессорный модуль обеспечивает запоминание состояния определенных входов и выходов МП, отключает МП от системы-прототипа и использует его в качестве процессора процессорного модуля внутрисхемного эмулятора для выполнения программ, хранящихся в ПЗУ отслеживающего модуля. Эти программы управляют процессом обмена информацией между внутрисхемным эмулятором и комплексом.

Отслеживающий модуль (рис. 10.4) подключают к шинам *отладочного комплекса* и внутренней шине эмулятора. Основными структурными элементами модуля является *регистр команд*, *ОЗУ* емкостью 256 байт, *ПЗУ* емкостью 1 Кбайт, *компаратор*, *мультиплексор*.

Регистр команд внутрисхемного эмулятора служит для приема управляющих директив от процессора комплекса. *Регистр состояния* содержит информацию о результатах выполнения эмулятором директивы комплекса. Об окончании выполнения эмулятором директивы процессору комплекса сигнализирует *схема формирования запросов на прерывания*.

Одним из наиболее важных функциональных элементов модуля является быстродействующее *ОЗУ* емкостью 256 байт, 176 байт из которых служат для запоминания информации о 44 последних циклах эмуляции. Такой информацией для каждого машинного цикла является код управляющего слова, код команды (данных), формируемых на *шине данных*, код адреса, формируемого на *шине адреса*. Формирование циклической последовательности адресов для записи этой информации осуществляется с помощью специального *адресного генератора*. Подача соответствую-

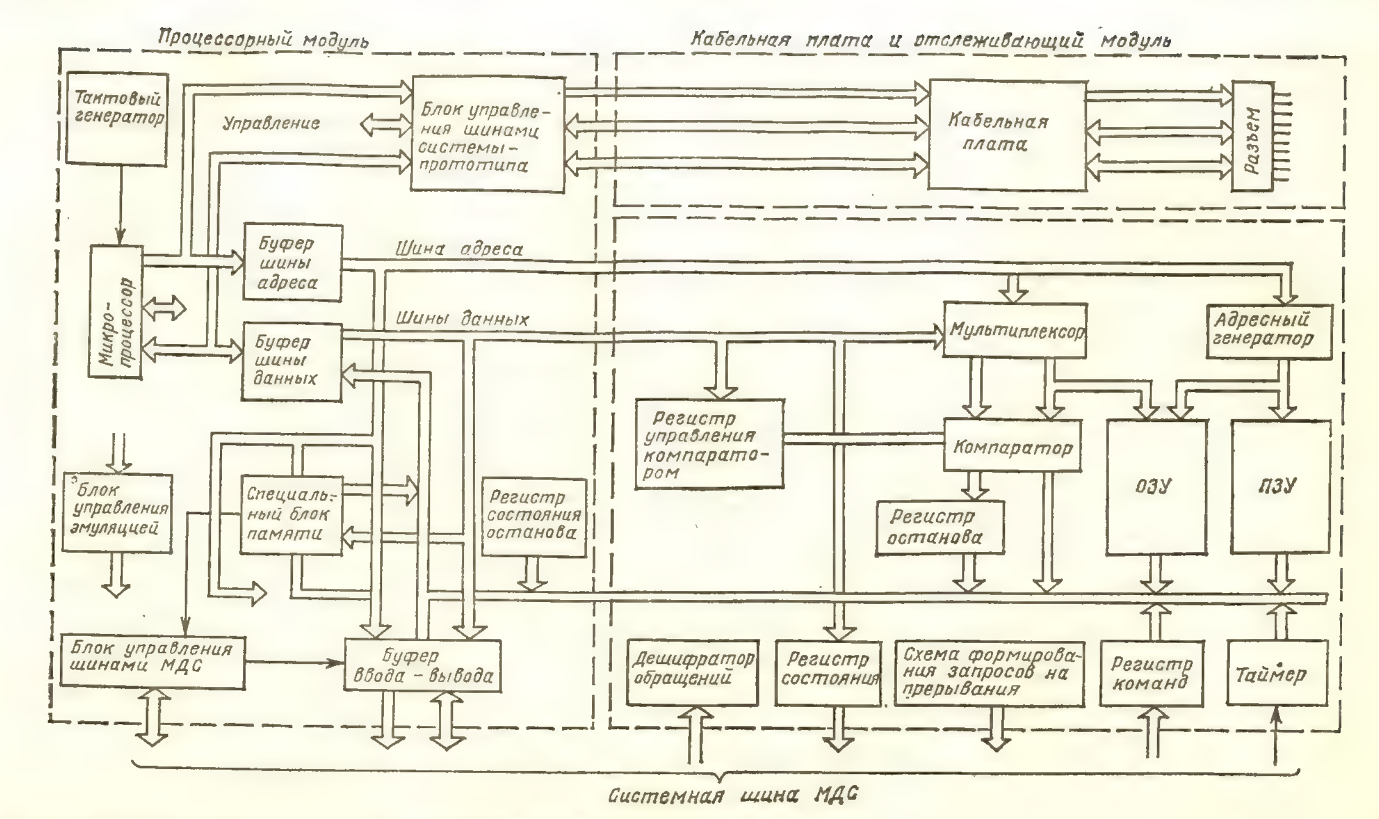


Рис. 10.4. Структурная схема внутрисхемного эмулятора ICE-80

щей информации одновременно с подачей адресов осуществляется с помощью *мультиплексора*.

Мультиплексор служит также для подачи на компаратор адресных и управляющих кодов, формируемых в каждом машинном цикле.

Компаратор представляет собой два 24-разрядных регистра, в каждый из которых загружается определенное пользователем условие останова эмуляции. В 24-разрядный формат входят 16-разрядный код адреса останова и 8-разрядный код управляющего слова. Во время эмуляции *компаратор* анализирует коды адресов и управляющих слов процессора, проходящие через *мультиплексор*. Совпадение этих кодов с кодом, записанным пользователем в любой из двух регистров *компаратора*, вызывает останов эмуляции по окончании выполнения текущей команды. Останов эмуляции может быть вызван и рядом других причин, например, попыткой эмулируемой программы обратиться к несуществующим ячейкам памяти или устройствам ввода-вывода, при переключении сигнала, контролируемого пользователем, в случае попытки обращения со стороны комплекса к эмулятору во время эмуляции либо, если во время эмуляции МП находится в состоянии ожидания или останова более 0,25 с.

Информация о причине останова заносится в *регистр останова*, содержимое которого может быть считано со стороны комплекса. В случае останова эмуляции процессор эмулятора переходит в командный режим, в котором он выполняет директивы, поступающие от комплекса.

В *отслеживающем модуле* расположено *ПЗУ* емкостью 1 Кбайт, в котором хранятся программы, позволяющие процессору эмулятора в командном режиме отвечать на директивы, поступающие через регистр команд со стороны комплекса. При этом 64 байта быстродействующего *ОЗУ отслеживающего модуля* используются в качестве рабочей области для программ, расположенных в *ПЗУ*. В состав отслеживающего модуля входит *таймер*, позволяющий установить чистое время выполнения эмулируемой программы.

Процессорный модуль включает МП 8080, который в зависимости от режима работы может подключаться либо к шинам системы прототипа, либо к внутренней шине комплекса. В *специальный блок памяти процессорного модуля* эмулятора заносится таблица соответствия адресов блоков памяти и внешних устройств, которые комплекс предоставляет системе-прототипу. В функции этого блока памяти входит анализ адресов блоков памяти и внешних устройств, с которыми работает эмулируемая программа. В случае обращения программы пользователя к запрещенной области адресов блоком формируется сигнал, вызывающий останов эмуляции. Блоком памяти идентифицируется также обращение программы пользователя к блокам памяти и внешним устройствам, физическое расположение которых назначено в комплексе. При

этом формируется сигнал для блока управления шинами комплекса, позволяющий подключить МП эмулятора к системной шине комплекса и сформировать все управляющие сигналы, необходимые для обращения этого МП к *ОЗУ* и внешним устройствам комплекса. Физические адреса *ОЗУ* и внешних устройств формируются в *буфере ввода-вывода* процессора эмулятора подстановкой физических номеров (адресов) блоков памяти, взятых из таблицы соответствия, вместо их логических номеров (адресов).

Блок управления шинами системы-прототипа обеспечивает полную имитацию присутствия МП в системе-прототипе во время эмуляции. Этот блок принимает и передает в систему-прототип все требуемые сигналы в каждом машинном цикле эмуляции, за исключением цикла ввода информации из блока памяти или внешнего устройства, физическое расположение которого назначено в отладочном комплексе. *Блок управления эмуляцией* в основном осуществляет управление процедурами инициализации.

Управление работой внутрисхемного эмулятора осуществляется по командам пользователя с помощью программы *ICE*-80, входящей в ДОС *ISIS*-II. Эта программа обеспечивает выполнение всех операций по отладке объектных программ в реальной микропроцессорной системе и по своим возможностям существенно превосходит монитор отладочного комплекса.

Универсальные аппаратно-программные отладочные комплексы. Ориентированы на различные МП с фиксированной системой команд и однокристальных микроЭВМ (КР580, К1810, К1814, К1816, К1820 и др.). Блоки ВСЭ и часть системного программного обеспечения являются сменными и предназначены для работы с конкретными МП [12].

Построение универсального отладочного комплекса с максимальными функциональными возможностями при наименьших затратах на разработку и умеренной стоимости изделия требуют создания аппаратных и программных средств по функционально-модульному принципу. Для этого необходимо следующее: ориентироваться на использование в качестве базового вычислительного комплекса массовых серийно выпускаемых микроЭВМ («Электроника 60» и ДВК); использовать сменные блоки эмуляторов и комплексы отладочных программ для каждого целевого МП; применять стандартные операционные системы и интерфейсы, унифицировать все аппаратные и программные средства по конструкции, интерфейсам, системной магистрали и системе команд; максимально реализовать функциональные возможности с помощью программных средств.

Вычислительный отладочный комплекс МИКРОСОТ является примером практического осуществления этих принципов.

Рассмотрим возможности комплекса по отладке аппаратно-программных средств пользователя на примере схемного эмулятора и управляющей программы для МПУ, построенных на базе микропроцессора КР580ИК80.

Схемный эмулятор содержит микропроцессор, тактовый генератор, память объемом до 64 Кбайт, а также 48-выводную соединительную вилку с разводкой, соответствующей назначению выводов МП, с помощью которой пользователь может подключить схемный эмулятор вместо МП и управлять процессом отладки с терминала МИКРОСОТ, используя аппаратные ресурсы как схемного эмулятора, так и МПУ. Так, например, можно объединить в систему тактовый генератор, порты ввода-вывода, находящиеся на плате разрабатываемого МПУ, и память схемного эмулятора, т. е. имеется возможность постепенного, поблочного наращивания аппаратных ресурсов МПУ.

Схемный эмулятор позволяет останавливать работу при появлении на адресной шине заданного адреса (останов по контрольным точкам, число которых ограничено лишь объемом памяти схемного эмулятора), после выполнения очередной команды (работа в пошаговом режиме), по окончании очередного машинного цикла процессора (работа в поцикловом режиме).

Информация на шинах МП в момент останова запоминается и может быть использована для тестирования МПУ статическими сигналами при отладке аппаратной части с применением различной контрольно-измерительной аппаратуры.

Аппаратные средства схемного эмулятора функционируют в соответствии с управляющей программой ОС ДВК, построенной по принципу диалога с оператором. Ниже приведены команды управляющей программы и их функциональное назначение:

BOOT — осуществляет загрузку отлаживаемой программы в кодах микропроцессора с гибкого магнитного диска или клавиатуры в память эмулятора. В первом случае программа должна быть предварительно оттранслирована с помощью кросс-ассемблера, входящего в состав комплекса отладочных программ;

CONF — изменяет конфигурацию аппаратных средств системы;

CR — устанавливает поцикловый режим работы процессора;

CYKL — реализует поциклическое выполнение заданного участка отлаживаемой программы;

GO — запускает отлаживаемую программу с произвольного адреса;

HALT — прекращает процесс эмуляции при обнаружении машинного цикла заданного типа;

KT — устанавливает контрольные точки в заданные адреса памяти;

LOAD — загружает информацию в ОЗУ МПУ;

MEM — изменяет содержимое любой ячейки памяти эмулятора;

NONSTOP — назначает непрерывный режим работы процессора, отменяя пошаговой и поцикловый режимы и контрольные точки;

P — продолжает выполнение отлаживаемой программы с текущего адреса;

R — изменяет содержимое выбранного регистра общего назначения центрального процессора;

RES — осуществляет начальный сброс процессора, а также обнуление содержимого регистров общего назначения, указателя стека и битов условий;

SHOW — индицирует конфигурации технических средств о режиме эмуляции и состоянии процессора в текущий момент времени;

STEP — устанавливает пошаговый режим работы процессора;

STOP — останавливает выполнение отлаживаемой программы в непрерывном режиме;

TRACE — индицирует значение 130 точек трассы отлаживаемой программы;

TYPE — индицирует в виде таблицы содержимое памяти, включенной в систему в текущий момент времени.

Помимо основных функций отладки реализован ряд сервисных возможностей: индикация мнемоники выполненной команды при работе в пошаговом режиме, возможность для пользователя работать в восьмеричной или шестнадцатеричной системе счисления и др.

Архитектура комплекса в целом и эмуляторов в частности предусматривает использование последних в автономном режиме без применения базового вычислительного комплекса. В этом случае управление работой блока эмулятора осуществляется специальным блоком через интерфейс связи после переключения эмулятора от ДВК к блоку управления. Такой принцип построения блоков эмулятора и управления делает их в значительной степени унифицированными: один блок эмулятора может быть использован и в составе комплекса и в автономном режиме; с другой стороны, для всех блоков эмулятора в автономном режиме используется один блок управления. Унифицированный корпус позволяет помещать блок эмулятора отдельно, а также совместно с блоком управления.

Перспективы развития универсального отладочного комплекса МИКРОСОТ связаны с расширением номенклатуры блоков эмулятора и комплексов отладочных программ для новых типов целевых МП, включением кросс-трансляторов с языков высокого уровня (Паскаля, СИ) в состав комплексов отладочных программ; разработкой универсального функционально-модульного схемного эмулятора с расширенными функциональными возможностями и сменных блоков эмуляции МП; использованием в качестве базового вычислительного комплекса перспективных моделей персональных ЭВМ и адаптации для них уже разработанных прикладных программных и аппаратных средств.

Рассмотрим схемный эмулятор МП К1810ВМ86 [36]. Архитектурные особенности построения блока эмулятора определяются

особенностями самого МП. Возможность программной реализации основных режимов отладки (пошагового, трассировки и режима контрольных точек) позволяет значительно сократить аппаратные затраты, поскольку достаточно организовать прямой доступ к памяти блока эмулятора ведущей микроЭВМ и подготовить соответствующую программу монитора.

В схемный эмулятор входят три регистра: команд, адреса и данных. Интерфейс ввода-вывода обеспечивает передачу управляющих команд и информации от ведущей микроЭВМ в регистры в зависимости от кода передаваемой информации.

Команда Установка ПДП передает управление памятью блока эмулятора ведущей микроЭВМ, команда Сброс ПДП — обратно центральному процессору К1810ВМ86. Команда RESET производит пуск программы пользователя, записанной в память МПУ с начального адреса FFFFOH (инициализация), команда TEST служит для возобновления выполнения программы после остановки ее по контрольной точке или шагу. Команда Пуск записи в памяти следа осуществляет установку режима контроля сигналов обмена МП с системой пользователя.

ОЗУ обеспечивает запоминание программы пользователя, переменных монитора, контрольных точек останова, состояния пошагового режима, трассировки, ПЗУ — хранение программы монитора, ОЗУ следа — запоминание 1024 состояний на шине центрального процессора, предшествующих точке останова.

Алгоритм управления программой следующий:

1. Передача управления командой Сброс ПДП центральному процессору блока эмулятора и инициализация системы командой RESET.

2. Захват памяти блока эмулятора командой Установка ПДП, обмен информацией между памятью блока эмулятора и управляющей микроЭВМ (загрузка ОЗУ программой пользователя).

3. Запуск командами Сброс ПДП и TEST программы пользователя.

4. Захват памяти блока эмулятора командой Установка ПДП и распечатка содержимого рабочих регистров ЦП на экране монитора.

Последующий пуск программы пользователя производится аналогично описанному выше. Управляющая программа позволяет модифицировать значения рабочих регистров, ячеек памяти, считывать ОЗУ следа по команде ПУСК ЗАПИСИ В ОЗУ СЛЕДА и оценивать состояние всех шин и управляющих сигналов ЦП за 1024 машинных цикла.

Управляющая программа для схемного эмулятора МП К1810ВМ86 создана в рамках комплекса отладочных программ МИКРОСОТ.

К преимуществам схемного эмулятора, используемого с ДВК-2М, относятся: ускоренный процесс отладки; применение языков высокого уровня; хранение программ на гибком магнит-

ном диске; вывод на экран монитора начальных, конечных, любых промежуточных состояний отлаживаемой программы; использование ОЗУ следа для анализа состояний всех шин центрального процессора до его останова; возможность дальнейшего расширения функций схемного эмулятора посредством подключения к его памяти следа логического анализатора.

10.3. СИСТЕМЫ ПОДДЕРЖКИ ПРОЕКТИРОВАНИЯ МИКРОПРОЦЕССОРНОЙ АППАРАТУРЫ НА ОСНОВЕ МИКРОПРОГРАММИРУЕМЫХ СЕКЦИОНИРОВАННЫХ МИКРОПРОЦЕССОРОВ

Разработка и отладка аппаратуры (устройств и систем) на основе микропрограммируемых МПК БИС существенно сложнее, чем в случае использования однокристальных микропроцессоров.

При разработке микропрограммного обеспечения следует учитывать следующие его особенности.

1. Связь микропрограммного обеспечения с техническими средствами микропроцессорных систем сильнее, чем в случае программного обеспечения.

При написании микропрограмм разработчик должен детально представлять функционирование и временные параметры работы технических средств.

2. Форматы и разрядность микрокоманды меняется от одной разработки к другой.

3. При микропрограммировании в основном используют языки типа ассемблера, что обусловлено высокими требованиями к эффективности получаемых микропрограмм.

4. Организация отладки микропрограмм до изготовления макета технических средств затруднена в связи с широким диапазоном микроархитектуры разрабатываемой микропроцессорной аппаратуры.

5. Для некоторых типов микропрограммируемых МПК БИС существуют сложные методы адресации микрокоманд. Учет таких методов усложняет микропрограммирование. Средства же автоматизации адресации микрокоманд не всегда доступны (иногда не существуют).

Все это делает особенно важным разработку и применение систем поддержки проектирования для микропрограммируемой микропроцессорной аппаратуры.

10.3.1. Программные (кросс) системы поддержки проектирования микропрограммируемой аппаратуры

Поскольку программное моделирование микропрограммируемой аппаратуры является достаточно трудоемкой задачей, в программных системах поддержки значительное место занимают системы без моделирования технических средств, позволяю-

щие автоматизировать написание и трансляцию микропрограммного обеспечения в объектные коды. Рассмотрим некоторые отечественные и зарубежные программные (кросс) системы поддержки проектирования микропрограммируемой аппаратуры без моделирования технических средств.

Система CROMIS. Система перекрестного микропрограммирования фирмы «Интел» помогает пользователю МПК серии 3000 в разработке и реализации микропрограмм [63].

Система *CROMIS* поставляется пользователю в форме двух программ, написанных на исходном языке Фортран-IV, *XMAS*

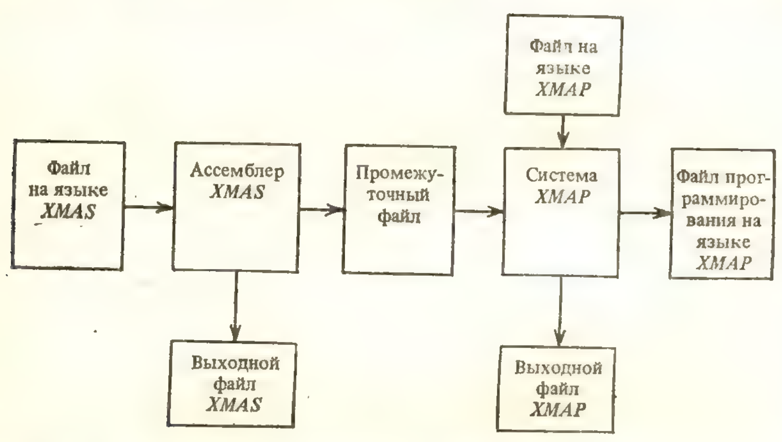


Рис. 10.5. Структура кросс-системы CROMIS

(перекрестный микроассемблер) и *XMAP* (генератор файла программирования ПЗУ).

Основная функция системы *CROMIS* заключается в преобразовании микропрограммы на языке *XMAS* в формат, подходящий для программирования индивидуальных ПЗУ и программируемых устройств, которые будут служить в качестве физической микропрограммной памяти. Эта функция связана с двумя фазами: фазой ассемблирования программы, написанной на языке *XMAS* и фазой выработки файла программирования ПЗУ.

Функциональные отношения между программами *XMAS* и *XMAP* представлены на рис. 10.5.

Система *XMAS* определяет символический, машинно-ориентированный язык микропрограммирования (язык *XMAS*). Язык *XMAS* учитывает специфику микропрограммных характеристик вычислительных элементов схем серии 3000.

Язык включает:

полный набор мнемонических обозначений для представлений функций центрального обрабатывающего элемента и микропрограммного устройства управления;

ограниченную символическую адресацию для упрощения представления микропрограммных последовательностей;

механизм для определения мнемоники, описывающий функции, управляемые полями, определенными пользователем;

специальное математическое обеспечение для поля слова микроинструкции, которое формирует сигналы на шине K центрального обрабатывающего элемента;

специальное обеспечение для адресации микропрограммной памяти объемом, превышающим 512 слов.

XMAS — язык со свободным форматом.

Универсальность языка *XMAS* в возможности его расширения. Из-за большого разнообразия конфигураций, которые возможны при использовании элементов серии 3000, конечный формат слова микроинструкции нельзя строить на языке *XMAS*. Следовательно, язык *XMAS* обеспечивает механизм для описания расширения слова микроинструкции в форме полей, определяемых пользователем.

Во время ассемблирования система *XMAS* читает записи из файла на исходном языке. Эти записи содержат операторы языка управления и операторы языка *XMAS*. Строго говоря, операторы управления не составляют части языка *XMAS*. Они задают форматы ввода-вывода, выбирают файлы и задают другие параметры.

Программа *XMAS* вырабатывает два файла во время ассемблирования: файл распечатки и файл микрокода.

Файл распечатки может содержать операторы файла на исходном языке, картотеку перекрестных ссылок, графическое представление образа микропрограммой памяти, вырабатываемого программой *XMAS*.

Система *XMAS* формирует промежуточный двоичный файл микрокода под управлением операторов на исходном языке. Для каждого описательного оператора в программе на языке *XMAS* система *XMAS* выводит в файл микрокода полное описание заданного слова микроинструкции.

Программа *XMAP* обрабатывает файл микрокода и формирует файл программирования ПЗУ или программируемых ЗУ.

Роль языка *XMAP* сводится к отображению последовательности битов в файле микрокода в заданной ячейке ПЗУ или программируемой памяти. Язык *XMAP* используется для описания организации компонент микропрограммной памяти для выполнения детальной процедуры отображения.

Во время выработки файла программирования ПЗУ программа *XMAP* читает записи из исходного файла на языке *XMAP* (см. рис. 10.5), который содержит операторы на языке управления и операторы отображения ПЗУ на языке *XMAP*.

При обработке программы *XMAP* система выдает: распечатку исходных операторов на языке *XMAP* и двоичную распечатку для каждого ПЗУ, задаваемого оператором отображения ПЗУ;

сводку по программе на языке *XMAP*. Пользователь выбирает нужную ему информацию с помощью языка управления.

Система SUMA — это система программирования на машинно-независимом языке повышенного уровня, разработанная в целях поддержки микропрограммирования для горизонтальных микроархитектур [62]. Она может быть легко применена для целевых машин самой различной архитектуры.

Система включает две различные, но связанные части:

описание целевой машины; описание алгоритма микропрограммы, который должен быть переведен для выполнения на этой машине.

Код на выходе компилятора служит исходным текстом для микроассемблера.

Небольшие изменения в архитектуре целевой машины не требуют больших затрат на переписывание микропрограммы, как в случае применения ассемблера. Потребуется только изменение представления самих операций, а не каждого случая их применения.

Если совершенствуется или модифицируется аппаратное обеспечение целевой машины, можно просто изменить описатели машины и(или) алгоритмы и перекомпилировать микропрограмму. Эти переделки не требуют от пользователя изменения компилятора или изучения через листинг микроассемблера исходного кода, для того чтобы искать в каждом случае определенные величины полей.

Система КРАС. Эта система ориентирована на автоматизацию микропрограммирования МПУ на базе МПК БИС серий КР1802 и К1804. При ее разработке была сделана попытка создать систему, которая находилась бы на достаточно высоком уровне по отношению к применяемым в мире, и в то же время авторы не хотели слепо копировать какой бы то ни было зарубежный аналог, обеспечив требование пользователей собственной организации.

Существенным этапом работы был выбор языка. Основными критериями выбора были: простота языка, его направленность на неподготовленного пользователя, специалиста по разработке микропрограммируемой аппаратуры; обеспечение полноты автоматизации программирования для всех выполняемых разработчиком операций; возможность использования языка для широкого круга микропрограммируемых МПК; распространенность языка, обеспечивающая переносимость и единое понимание микропрограмм.

В качестве базового языка был принят язык *AMDASM* со следующими изменениями и дополнениями.

Введены операторы постпроцессора, определяющие характер информации на выходе транслятора, формат носителя информации, процедуры работы с получаемой на выходе матрицей микрослов. Такими операторами являются:

PPZU — управляющий получением носителя информации в форме, пригодной для программирования ППЗУ;

OZU — то же для формата, пригодного для занесения матрицы микрослов в ОЗУ;

IZM — определяющий характер изменения выходной матрицы микрослов.

Введена возможность работы с русскими обозначениями во всех местах программы.

Введен оператор *ZNX*, позволяющий определять значения, которыми заполняются поля микрослов, обозначенные X.

Изменены некоторые операции с модификаторами и атрибутами.

Язык микропрограммирования, являющийся входным для системы КРАС, назван ЯП1804. Программа на языке ЯП1804 условно делится на 3 фазы (в *AMDASM* программа делится на две фазы):

первая фаза — фаза определения, служит для задания форматов микрокоманд и определения функций полей;

вторая — рабочая фаза, служит для выработки операций по заполнению форматов микрокоманд соответствующими требованию пользователей именами функций, работы со счетчиком;

третья — фаза вывода, предназначена для подготовки информации вывода в удобном для пользователя виде.

Каждая фаза имеет присущие только ей специфические слова — определения и заканчивается ключевым словом *END*.

Транслятор КРАС работает в среде ОС ЕС и написан на языке системы РТК (P-TPAN). Объем транслятора ≈110 Кбайт. Входной информацией транслятору служит язык ЯП1804. Выходной информацией транслятора является распечатка исходной информации и выходной матрицы, а также перфоленты.

В зависимости от этапа работы над программой пользователь может работать в любой из фаз трансляции своей программы. Пользователю допускается писать программы на языке ЯП1804 объемом до 50 Кбайт, что эквивалентно 2000 микрокоманд. Предполагаются два режима работы: пакетный и диалоговый (на системе ЕС-7970).

Дальнейшее развитие системы предполагает следующее: введение системы постраничного вывода информации; учет особенностей работы нескольких МПК; повышение отладочных возможностей и сервиса.

Система КРОССМПК. По техническим данным система КРОССМПК напоминает систему *CROMIS*, однако, имеет бóльшие возможности, а именно: микропрограммируемая аппаратура может быть построена на базе МПК серии К589 и на базе любого другого микропрограммируемого МПК БИС; система обладает более сильными, чем в *CROMIS*, средствами описания синтаксиса и семантики генерируемого микроассемблера.

Основной функцией КРОССМПК является преобразование микропрограмм, написанных на входном языке *INLA*, в форму, пригодную для программирования различных устройств ПЗУ

и ППЗУ. Эта функция реализуется в два этапа: ассемблирование программы на языке *INLA* и генерация файла для программирования ПЗУ.

Структурная схема системы представлена на рис. 10.6. КРОССМПК разбита на три основных модуля и ряд вспомогательных программ. Модуль *TABL* предназначен для формирова-

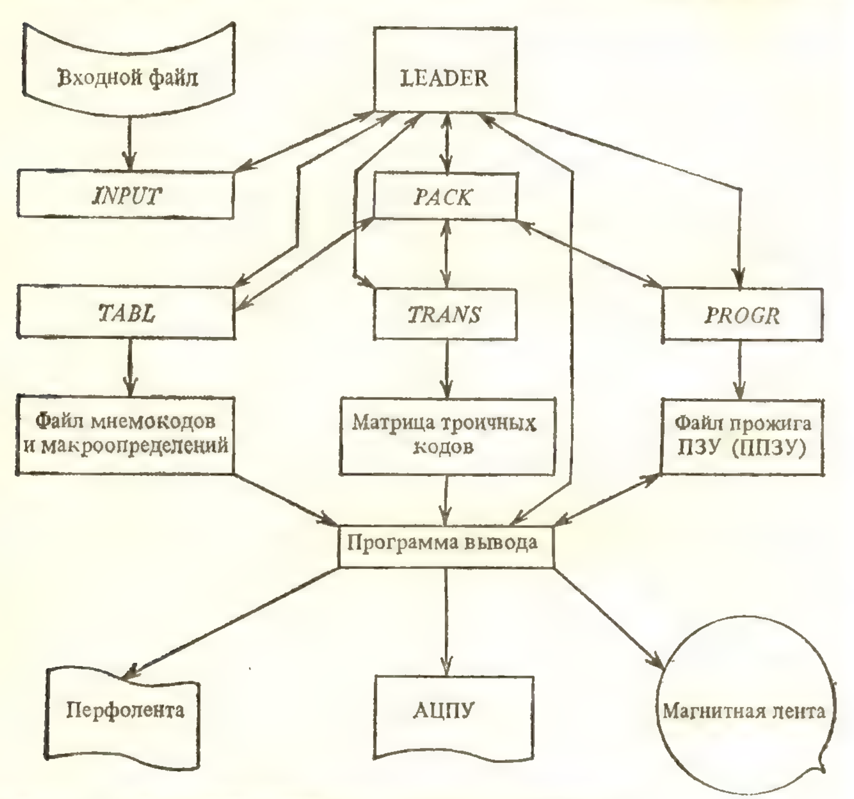


Рис. 10.6. Структурная схема кросс-системы КРОССМПК

ния таблиц мнемокодов, модуль *TRANS* — для трансляции исходных микропрограмм в матрицу троичных кодов. Для представления значений разрядов полей используются символы 0,1 и X (X — для безразличных значений разрядов полей). Модуль *PROGR* служит для формирования *файла «прожига» ПЗУ*.

Программа INPUT предназначена для выделения из *входного файла* лексических единиц языка. Программа *PACK* предназначена для упаковки идентификаторов и чисел, выделенных программой *INPUT*. Эта программа используется модулями системы для компактной записи обработанной информации. Программа *LEADER* является управляющей программой системы

и обеспечивает распределение управления между программами и модулями КРОССМПК.

Система реализована на языке ФОРТРАН-IV и может быть поставлена на любой ЭВМ, имеющей транслятор с этого языка и емкость оперативной памяти не менее 256 Кбайт. Система рассчитана на работу в пакетном режиме. Время трансляции 512 микрокоманд 12 мин.

Система MICRO. Язык микропрограммирования *MICRO* имеет уровень ассемблера. Транслятор реализован на микроЭВМ «Электроника-60». В процессе трансляции генерируются два файла — традиционный листинг и загрузочный файл. Имеет средства связывания раздельно оттранслированных микропрограмм посредством аппарата глобальных меток. Настройка на формат микрокоманды не формализована.

Кросс-система автоматизации микропрограммирования MICROP. Система состоит из двух частей: ЯМП с соответствующим транслятором и архива микропрограмм с независимыми средствами доступа [20].

Использование архива позволяет накапливать готовые микрокоды и выдавать технологическую документацию независимо от транслятора. Средства доступа к архиву ориентированы на постоянное расширение. Технические данные системы, которая реализована на базе ОС ЕС ЭВМ, следующие: длина микрокоманды — до 200 разрядов; емкость памяти микрокоманд — до 2 Кслов; время трансляции одной микрокоманды длиной 100 разрядов — около 0,1 с; длина поля — до 8 разрядов.

ЯМП системы является перенастраиваемым. Перенастройка может быть осуществлена на двух уровнях — на новое устройство и на новый микропроцессорный комплект. Настройка на новый комплект требует около 1 человеко-месяца работы. При настройке на новое устройство участия разработчиков не требуется. Имеются варианты системы для К1804, КР1802, К589. Система ориентирована на пакетный режим работы.

Кросс-системы с программным моделированием. Наряду с кросс-системами поддержки проектирования микропрограммируемой микропроцессорной аппаратуры без моделирования технических средств, несмотря на трудности, широкое распространение находят кросс-системы с использованием программной модели технических средств.

Совместная проверка и отладка технических средств и микропрограммного обеспечения на макете является обязательной при любой методике проектирования. Однако такая проверка и отладка возможна только после изготовления макета технических средств. Кроме того, организация отладочного режима на макете даже с применением мощных аппаратно-программных отладочных комплексов не всегда удовлетворяет пользователя (например, для очень «изощренных» микроархитектур). Отладка же микропрограмм на программной модели может производиться на ранней

стадии разработки до изготовления опытного образца. При этом может быть предусмотрен широкий диапазон отладочных операций, некоторые из которых могут учитывать конкретные микроархитектуры.

Система IMPULSE. Является одной из наиболее известных за рубежом. Основные усилия ее создателей были направлены на

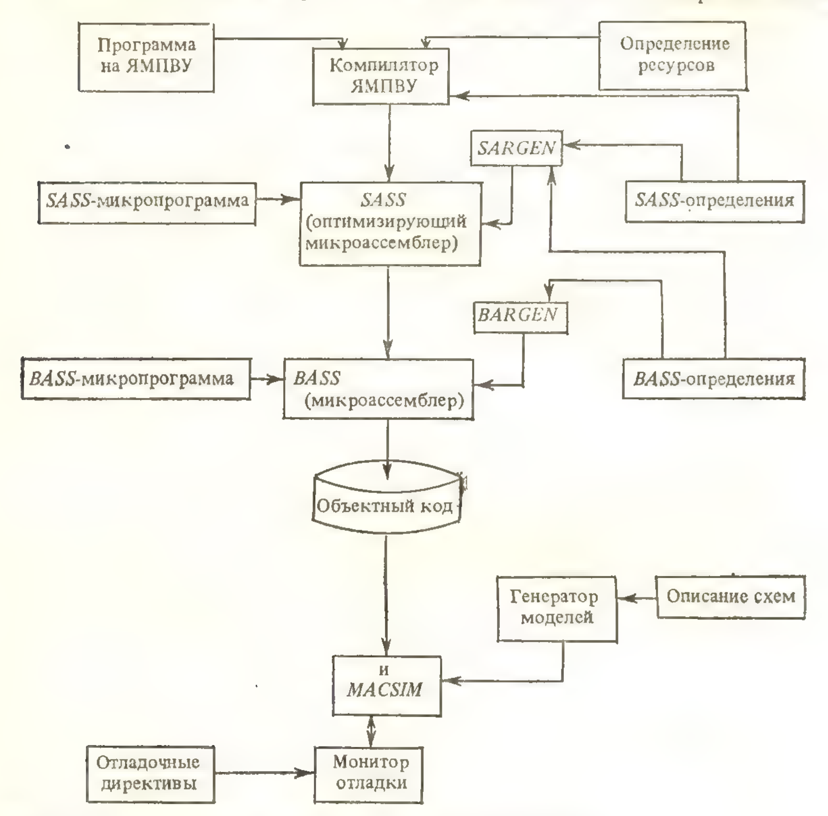


Рис. 10.7. Структурная схема кросс-системы IMPULSE

повышение уровня ЯМП. Методы, реализованные при разработке, базируются на серии фундаментальных исследований в области теории микропрограммирования. Однако как и любая чисто кроссовая система, *IMPULSE* не может обеспечить полную отладку микропрограмм [64].

Конфигурация системы показана на рис. 10.7. Система подразделяется на две большие части: генерация микропрограмм и отладка. Программа *MACSIM* обеспечивает моделирование аппаратуры на различных уровнях.

Генератор микропрограмм подразделяется на два уровня. Теоретически допускается запись микропрограмм на любом из

трех уровней (при переходе от уровня к уровню снизу вверх уменьшается зависимость формы записи микропрограмм от конкретной аппаратуры), однако реально реализованы лишь два нижних. Метод напоминает методику, реализованную в отечественной системе *НЯМУНАС-91* — при переходе от верхних уровней к нижнему, уровню традиционного микроассемблера, требуется вводить в транслятор все больше информации об аппаратуре.

Верхний уровень представляет собой ЯМП высокого уровня.

Ниже расположен уровень, реализуемый программой *SASS*, которая позволяет записывать микропрограммы в виде потока микроопераций, не сгруппированных в микрокоманды. Оптимизирующий ассемблер сам формирует микроинструкции.

Программа *BASS* реализует метамикроассемблер с традиционными свойствами. Запись микропрограммы на входном языке *BASS* может иметь очень наглядный вид, так как требуется довольно детальное описание при настройке как исходного формата, так и форматов генерируемых микрокоманд. Вместе с тем описания хорошо формализованы и вполне могут быть выполнены непрофессионалом.

Система реализована на языке Фортран-IV на машине с системой команд *IBM* и использовалась в нескольких разработках микропрограммируемой аппаратуры, в том числе и на основе МПК АМ2900.

В настоящее время ведутся интенсивные работы по реализации ЯМП высокого уровня для устройств, построенных на основе секционируемых микропроцессоров.

Система DEFASM — одно из наиболее серьезных достижений в этой области. Общая организация системы представлена на рис. 10.8.

Основными частями системы являются: метаассемблер *DEFASM*; компилятор, который по тексту на языке ПЛ/М генерирует объектный код микропрограммы; система символического исполнения микропрограмм на основе языка регистровых передач *RTS1a*.

Заметим, что в отличие от большинства подобных систем компилятор и микроассемблер работают независимо друг от друга.

Каждая часть системы разрабатывалась независимо от других.

Метаассемблер *DEFASM* — один из наиболее сильных языков подобного типа. Транслятор реализован на Фортран-IV и может работать на многих ЭВМ.

Компилятор с ЯМП реализован на языке Паскаль. Работа над компилятором продолжается в направлении оптимизации распределения ресурсов.

Разработчики системы предъявляли следующие требования к подсистеме моделирования: проверка логики; временной анализ (доступность шин, конфликты по ресурсам и т. д.); проверка корректности использования физических устройств.

Требовалось осуществлять моделирование на разных уровнях представления процессов. Язык регистровых передач *RTS1a* удовлетворяет всем названным требованиям.

Система ГНОМ/ФЛОТ. Универсальный микроассемблер ГНОМ (генератор объектного микрокода) [33] реализован на миниЭВМ СМ-4 и имеет следующие особенности. Максимальная разрядность слова 256 бит, причем микрокоманда может содержать

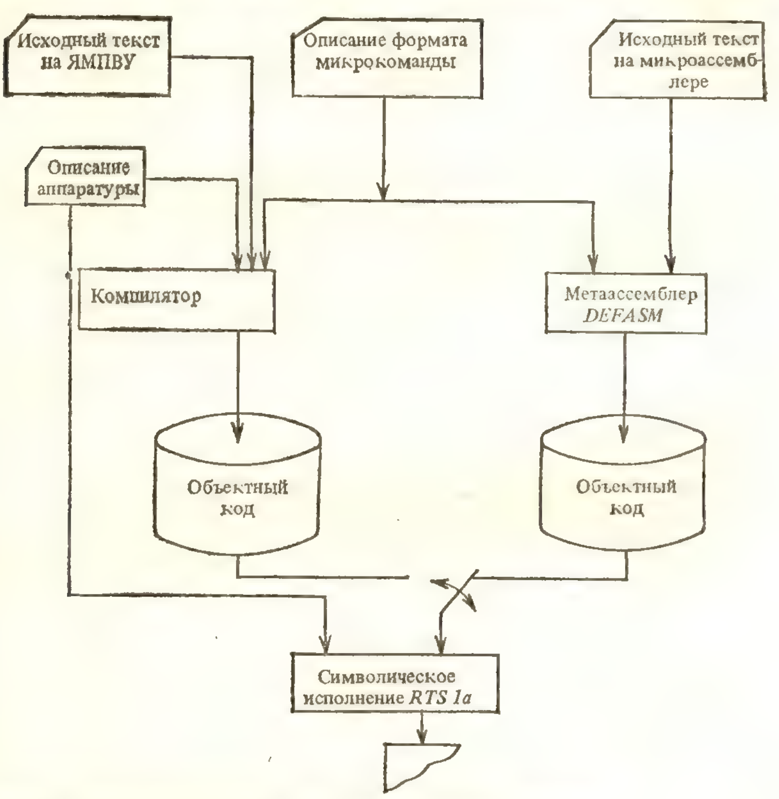


Рис. 10.8. Структурная схема кросс-системы DEFASM

от одного до семи слов. Допускаются любые идентификаторы, начинающиеся с латинской буквы и ряда специальных символов длиной до 128 символов. При настройке микроассемблера и задании символических микрокоманд возможно использование выражений, состоящих из идентификаторов, числовых констант, арифметических и логических операций и круглых скобок. При выполнении операций операнды считаются целыми числами без знака, имеющими разрядность, равную разрядности слова микрокоманды. Язык микроассемблера содержит набор псевдокоманд, позволяющих присваивать числовым значениям мнемонические

обозначения, резервировать участки памяти микрокоманд, заносить в ПЗУ числовые константы, устанавливать в счетчике размещения микрокоманд заданное значение, обозначать набор полей и субполей микрокоманды с их заданными значениями одним идентификатором и использовать его в дальнейшем. Предусмотрено автоматическое последовательное размещение микрокоманд в ПЗУ, имеется также возможность задавать численное значение адреса любой микрокоманды. Универсальный микроассемблер ГНОМ может использоваться при разработке микропрограмм для микропрограммируемых МПК БИС любой серии.

Следующим этапом разработки микропрограмм является их отладка на программной модели технических средств.

Система функционально-логической отладки микропроцессорных устройств ФЛОТ позволяет отлаживать не только микропрограммное обеспечение, но и функционирование технических средств. Базой для получения модели является библиотека функциональных описаний микропроцессорных БИС и автоматическое формирование модели технических средств в соответствии с заданной схемой соединения. Модели БИС должны адекватно описывать логические сигналы на выводах БИС и изменение содержимого микропрограммно-доступных и управляющих регистров БИС. В соответствии с заданной схемой ЭВМ должна автоматически формировать модель технических средств. Процесс отладки микропрограмм в этом случае состоит из следующих этапов:

задания схемы соединения БИС, образующих технические средства микропроцессорной системы;

автоматического формирования программной модели технических средств, адекватно эмулирующей микроархитектуру технических средств и логические сигналы на выводах всех БИС;

задания на микроассемблере отлаживаемой микропрограммы, трансляции ее в двоичный микрокод и представления последнего либо в виде состояний ячеек управляющего ЗУ, либо в виде последовательности логических сигналов, поступающих с блока микропрограммного управления; задания начального состояния внутренних микропрограммно-доступных и управляющих регистров, а также логических сигналов связи микропроцессорной системы с внешними объектами;

указания разработчиком перечня выводимой на экран или печатающее устройство информации (состояния заданных сигналов, содержимое регистров) и условий вывода;

проведения моделирования и анализа результатов;

корректировки микропрограмм либо на уровне символических микрокоманд с повторной трансляцией, либо на уровне двоичного микрокода, а также в ряде случаев корректировки принципиальной схемы технических средств.

Основным недостатком такого метода являются большие затраты машинного времени в связи с подробным моделированием работы БИС.

Система ФЛОТ имеет следующие особенности.

1. Используемые модели БИС являются функциональными, т. е. подробно не отражают структуру БИС на уровне вентилей. Формируемая модель технических средств микропроцессорной системы является логической моделью, т. е. эмулирует кроме изменения содержимого микропрограммно-доступных и управляющих регистров логические сигналы во всех точках принципиальной схемы. Логические сигналы представлены в двузначной логике с учетом состояния с высоким выходным сопротивлением.

2. Предусмотрена возможность моделирования двунаправленных шин и линий микропроцессорных систем, а также объединения выводов БИС с использованием проводного И, проводного ИЛИ, объединения выводов в общую линию, при котором в каждый момент времени сигнал только на одном выходе может иметь активное значение.

3. Подсистема и язык разработки моделей БИС отделены от языка задания схемы соединения БИС и входных воздействий. Разработка моделей БИС проводится администратором системы, знакомым со всеми ее тонкостями. Управление моделированием осуществляется с помощью диалогового языка, команды которого можно разбить на следующие группы:

команды работы с моделью, обеспечивающие задание и модификацию схемы соединения БИС, а также запуска и останова моделирования;

команды задания входных воздействий и начальных значений, перечня и условий выдачи содержимого регистров и состояний логических сигналов как во время моделирования, так и во время останова;

универсальные команды программирования: присваивания, перехода, организации цикла, обращения к подпрограммам. Эти команды позволяют производить операции над значениями логических сигналов, содержимым регистров, целочисленными переменными и одномерными массивами.

Команды могут выполняться сразу после их ввода пользователем с терминала или быть сгруппированы в программу управления и выполнены при запуске моделирования. Использование таких программ позволяет пользователю заранее спланировать последовательность моделирования, которое затем осуществляется в автоматическом режиме, и организовать вывод результатов в требуемом виде.

При отладке микропроцессорной системы по частям отсутствующие части можно легко смоделировать функционально, используя язык управления.

4. Задание микропрограмм может осуществляться либо на микроассемблере ГНОМ, либо непосредственно в виде объектного микрокода с помощью языка управления.

5. По своей организации программа моделирования является управляемым интерпретатором, что позволяет легко модифици-

ровать моделируемую схему технических средств. При этом используется событийное синхронное моделирование или моделирование с равными задержками. При этом предполагается, что в промежутке между тактами или между изменениями сигналов на входах микропроцессорной системы все переходные процессы заканчиваются. Для контроля правильности принципиальной схемы на отсутствие состязаний предусмотрен подсчет и выдача по запросу пользователя количества изменений входных сигналов каждой БИС в одном такте моделирования. Подобная верификация временных диаграмм технических средств с учетом разброса задержек и ограничений на временные диаграммы входных сигналов каждой БИС должна осуществляться отдельно специальной программой логического моделирования с использованием многозначного представления сигналов.

Рассмотренный метод отладки микропрограммного обеспечения позволяет проводить его на ранней стадии проектирования микропроцессорных систем, построенных на базе секционных МПК БИС, сравнивать варианты построения микропроцессорных систем, а также проводить функциональный контроль схемы технических средств.

Система НЯМУНАС-91. Система автоматизации микропрограммирования состоит из ядра, инвариантного к технологической базе, и средств, ориентированных на конкретные микропрограммируемые микропроцессорные структуры [25]. Данная система автоматизированного проектирования включает следующие этапы:

написание исходных микропрограмм, не зависящих от применяемой микропроцессорной базы, на языке высокого уровня;

проверка правильности алгоритма исходных микропрограмм помощью моделирования на алгоритмическом уровне;

трансляция исходных микропрограмм в объектные микропрограммы на уровне микрокоманд, ориентированных на применение определенного микропроцессорного комплекта;

формирование полной имитационной модели структуры проектируемого устройства и проверка правильности функционирования объектных микропрограмм на созданной модели;

получение технической документации и отлаженных микропрограмм на ЭВМ.

На рис. 10.9 представлена схема автоматизированной отладки микропрограмм для цифровых микропрограммируемых устройств (ЦМУ). Для проверки правильности микропрограмм используется двухуровневая модель, позволяющая выявить алгоритмические ошибки при моделировании исходных микропрограмм, а логические ошибки — при моделировании на микрокомандном уровне. Начальная микропрограмма, написанная на языке ЯМПВУ МИКАЛУ, отлаживается на алгоритмическом уровне. В результате алгоритмического моделирования (*А Модель*) получаются тесты для проверки объектных кодов микропрограмм, базирую-

щихся на конкретной микропроцессорной структуре. Для моделирования объектных микропрограмм строится *М Модель*. Определены три основных уровня представления исходных данных: 1) алгоритмический; 2) описание структуры ЦМУ; 3) микрокомандный.

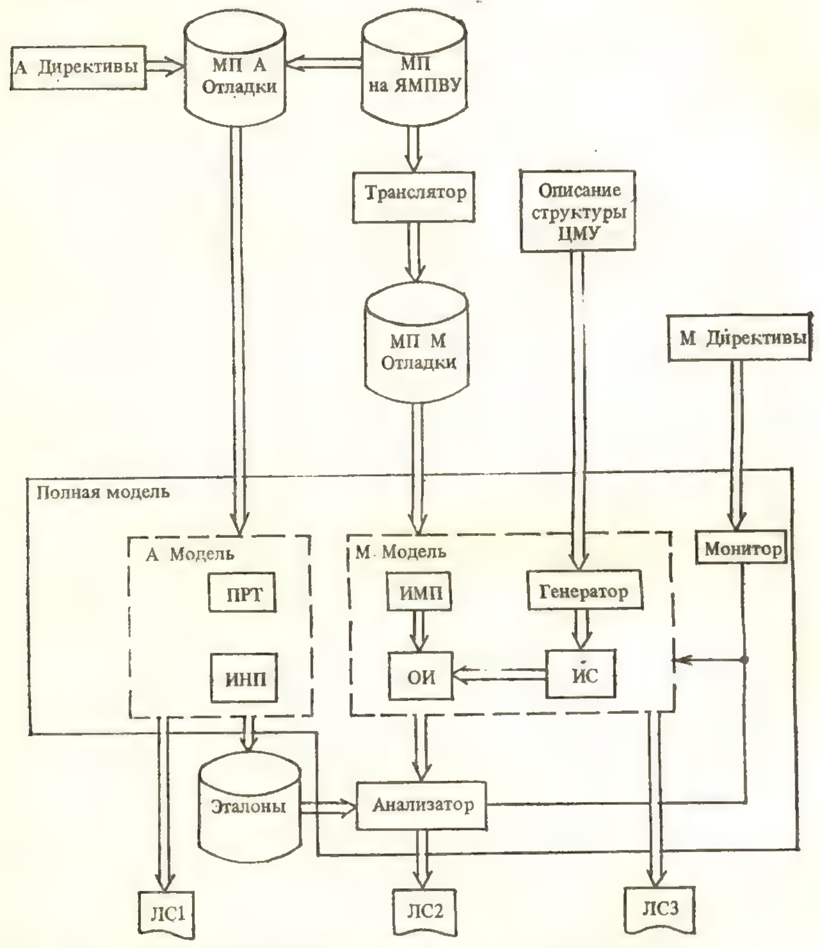


Рис. 10.9. Структурная схема кросс-системы НЯМУНАС-91

Такому представлению соответствуют и получаемые результаты автоматизированной отладки: 1) *ЛС1* — листинг сообщений моделирования на алгоритмическом уровне (выполняет *А Модель*); 2) *ЛС2* — листинг сообщений моделирования на микрокомандном уровне (выполняет *М Модель*); 3) *ЛС3* — листинг сопоставления результатов моделирования на обоих уровнях (выполняет *Анализатор*).

Обмен информацией между отдельными этапами моделирования происходит с помощью внешних носителей информации.

Для отладки исходных микропрограмм в текст вписываются управляющие директивы *А Директивы*, позволяющие совместно со средствами языка моделирования (ПЛ/1) исследовать различные ситуации при выполнении микропрограмм. Основными элементами *А Модели* являются предтранслятор *ПРТ* с ЯМПВУ в язык моделирования и интерпретатор *ИНП* внутреннего представления микропрограмм *МП А Отладки*.

При помощи директив трассировки определяются режимы печати значений переменных для дальнейшего моделирования. Операторы *ЯМПВУ* разрешают выполнить присваивание исходных значений, проверить условия вывода информации, окончания и повторения моделирования и другие алгоритмические действия. Операторы языка моделирования используются для вычисления значений математических функций, для вывода накоплений в процессе моделирования информации в редактированном виде.

Использование в качестве языка моделирования ПЛ/1 разрешает довольно просто реализовать отдельные программные модули *А Модели* (ЯМПВУ МИКАЛУ практически является подмножеством ПЛ/1) и облегчает связь между информационными массивами.

Для отладки *А Модели* предлагается следующий порядок. Сначала использовать выдачу значений отдельных переменных в ключевых местах микропрограммы, затем применять трассировку отдельных переменных при записи, при записи и чтении. И только для выявления особых ошибочных ситуаций использовать трассировку всех переменных. Информация трассировки накапливается в буфере, после заполнения которого выводится на печать.

Некоторым недостатком *А Модели* надо считать повторную предтрансляцию микропрограмм и копиляцию с ПЛ/1 при изменении отладочных действий.

Для отладки микропрограмм на микрокомандном уровне используется *М Модель*, состоящая из системных и структурных элементов. Под системными элементами понимаются имитаторы применяемого микропроцессорного комплекта, которые изменяются только при переходе на новую технологическую базу. Под структурными элементами подразумеваются модули, которые генерируются из языка, описывающего структуру проектируемого ЦМУ. Данный язык позволяет указывать параметры стандартных блоков (микропроцессорных секций) и способы их соединения. Для генерации структурных моделей — имитаторов структуры *ИС* имеется *Генератор*, который по описанию структуры ЦМУ формирует программные модули. Совокупность *ИС* и системных элементов — имитаторов микропроцессорной базы *ИМП* образуют объединенный имитатор *ОИ* проектируемого ЦМУ.

Для отладки микропрограмм на *М Модели* используются *М Директивы*, которые указывают режимы работы для *Монитора*, осуществляющего контроль процесса моделирования. Предусмотрена возможность моделирования микропрограмм с символическими адресами, т. е. физическая адресация выполняется один раз с логически отлаженными микропрограммами.

Процесс отладки ведется в диалоговом режиме. Анализируя листинг сообщений, проектировщик может изменять очередность задаваемых директив.

В режиме трассировки информация об изменяющихся значениях элементов моделируемого ЦМУ выводится в фиксированном формате.

Для проверки микропрограмм с Тестами, полученными во время работы А Модели, применяют *Анализатор*, сопоставляющий результаты выполнения микропрограмм с помощью директив в заданных точках.

М Модель данной системы ориентирована на микропроцессорный комплект К589. Настройка системы выполняется на МПК серии К583 и КР1802, намечается настройка на МПК серии К1804.

Моделирование на уровне микрокоманд информационно связано с результатами работы макроассемблера, также входящего в систему автоматизированного микропрограммирования [6]. Рассмотрим особенности макроассемблера *CROMAC*, разработанного для повышения эффективности проектирования микропрограммируемых устройств. На базе макрорасширений разработана функционально определяемая система операторов, с помощью которой можно описывать алгоритм функционирования проектируемого устройства в виде микропрограмм. Опознавание макровызовов по эталону позволяет создать систему макродействий, близкую по своей выразительности к операторам языков высокого уровня. Предусмотрена возможность транслировать микропрограммы без абсолютных адресов с символическими метками перехода и получать в результате матрицы двоичных кодов для управляющих полей микрокоманды и массивы (таблицы) символьных переходов. Результатом работы макроассемблера являются объектные микропрограммы.

Для настройки макроассемблера на определенную микропроцессорную базу используется язык, позволяющий проектировщику описать формат микрокоманд и мнемонику применяемых микроопераций. Настройка функционально определяемой системы операторов осуществляется программными средствами, выполненными на основе макрорасширения текста. При этом основной языковой конструкцией является макроопределение.

Данный макроассемблер разработан на языке ПЛ/1 ДОС ЕС ЭВМ.

10.3.2. Аппаратно-программные комплексы

Рассмотрим комплексы, наиболее известные в нашей стране и за рубежом.

Система 29. Система предназначена для поддержки проектирования микропрограммируемых систем и используется для разработки аппаратных и программно-аппаратных средств, а затем и для наладки макета системы. Инженеру-разработчику система предоставляет следующие возможности: интерактивное редактирование микрокодов и исходных файлов, ассемблирование микрокодов; ассемблер микропрограмм *AMDASM*/29; контроль состояния макета для отладки микропрограммируемой аппаратуры.

Система 29 представляет собой две отдельные системы в едином исполнении [43] — систему вспомогательного процессора и микропрограммируемую систему. Система вспомогательного процессора выполнена на основе микропроцессора *Am*9080A и обеспечивает связь пользователя с микропрограммируемой системой.

Система вспомогательного процессора содержит:

микропроцессор *Am*9080A для обеспечения управления системой;

два гибких диска для хранения операционной системы, включая ассемблер *AMDASM*/29;

плату памяти в виде оперативного запоминающего устройства на 32 Кбайт для работы процессора *Am*9080A;

последовательный интерфейс для связи с дисплеем пульта оператора, а также последовательный интерфейс *RS*-232 для связи с устройством печати и перфоратором. Возможна связь с программатором ППЗУ;

регистр страниц для хранения номера активной страницы.

Микропрограммируемая система содержит:

оперативную управляющую память на двух схемных платах, в которых используются микросхемы статических ОЗУ с МОП-структурой емкостью 4 Кбит. Это позволяет получить память микропрограмм общей емкостью 32 Кбит, либо 2048 слов по 128 бит, либо 4096 слов по 64 бит. Такая память позволяет производить отладку микропрограмм при пониженной частоте разрабатываемой системы. Возможно и применение памяти 1024×64 бит с временем выборки 50 нс на основе биполярных ОЗУ (выборка — 30 нс), позволяющей отлаживать микропрограммы на реальной частоте;

инструментальную плату. Эта плата реализует логику, необходимую вспомогательному процессору для непосредственного управления микропрограммируемой системой, и содержит средства управления синхронизацией — задает такие режимы, как останов, пошаговый, работа и прерывание на программной контрольной точке, а также управление прерываниями и переходами по адресу микропрограммы. Останов, пошаговый режим и работа (прогон) реализуются под управлением как программного обес-

печения вспомогательного процессора, так и органов лицевой панели, в то время как прерывание на контрольной точке происходит под управлением программных средств. Двенадцатиразрядный адрес контрольной точки (контрольного останова) загружается программным способом в регистр и сравнивается с 12-разрядным адресом оперативной управляющей памяти. Когда они совпадают, синхронизация микропрограммируемой системы останавливается и посылается запрос на прерывание в вспомогательный процессор.

Программным способом можно задавать адрес перехода, так что схема управления последовательностью микрокоманд начнет выполнение микропрограммы с этого адреса. Такая возможность позволяет задавать начальный адрес или переходить на диагностическую микропрограмму при отладке рабочей микропрограммы.

Принятая страничная структура обеспечивает приблизительно десятикратный выигрыш в производительности по сравнению с системой, в которой для работы с макетом используются контроллеры ввода-вывода. Этот выигрыш объясняется тем, что для доступа к микропрограммируемой системе применяются команды обращения к памяти, а не команды ввода-вывода. Такая структура предоставляет 256 индивидуальных страниц для выполнения различных функций в системе проектирования. Поскольку в самой системе занято только минимальное число листов, пользователь может при разработке устройств вводить собственные дополнительные схемы, что обеспечивает максимальную эффективность.

В системе предусмотрено трассирующее ОЗУ объемом 256 100-разрядных слов. Контрольные точки макета записываются в это ОЗУ с синхрочастотой работы макета (или с частотой дробной или кратной основной синхрочастоте).

При останове синхрочастоты (например, при останове по заданному адресу микрокоманды) последние 256 состояний этих контрольных точек могут отображаться на дисплее в формате, заданном пользователем.

Вместо того чтобы разрабатывать собственное микропрограммируемое управляющее устройство, пользователь может выбрать контроллер (построенный на базе БИС *Am*2900), выпускаемый для Системы 29. Для этого контроллера пользователь может написать свои микропрограммы применительно к конкретной разработке.

Программное обеспечение для Системы 29 включает четыре основные части: систему ввода-вывода, дисковую операционную систему, процессор пультовых директив и область сменных программ.

Дисковая операционная система *AMDOS*/29 позволяет открывать, закрывать, переименовывать, считывать, записывать файлы на диск или искать по имени. При использовании этой системы файлов можно хранить большое количество различных программ как в исходной форме, так и в форме, обрабатываемой

машиной. Дополнительно дисковой операционной системой предоставляются все операции, необходимые для доступа к дисководам и для связи с такими стандартными периферийными устройствами, как телетайп, дисплей, устройство ввода-вывода перфоленты, программатор и другие периферийные устройства.

Процессор пультовых директив считывает входные сообщения с пульта и соответственно этим директивам распечатывает листинги каталога файлов, содержимое различных файлов и управляет работой различных стандартных программ, поставляемых с *AMDOS*/29.

В области сменных программ размещаются программы, загружаемые из дисковой памяти для выполнения системой. Некоторые из них поставляются вместе с дисковой операционной системой. Типичными являются следующие программы:

ассемблер микропроцессора *Am*9080A;

определение состояния — выдает статистическую информацию об отдельных файлах на диске;

отладчик — позволяет отлаживать программы для микропроцессора *Am*9080A;

программа взаимного периферийного обмена — осуществляет операции промежуточного преобразования при обмене между периферийными устройствами;

редактор — загружает и выполняет редактирование текстов для *Am*9080A;

группировка — обеспечивает групповую обработку различных команд дисковой операционной системы.

Помимо этих программ предусмотрены мощные средства генерации микропрограмм. Они включают:

AMDASM/29 — двухпроходной ассемблер микропрограмм, используемый для преобразования установленного набора форматов микрокоманд в машинный язык (микрокод);

AMPROM/29 — программа пост-обработки, которая позволяет пользователю выдавать свой двоичный объектный код в форме, соответствующей организации ППЗУ;

AMSCPAM/29 — программа пост-обработки, позволяющая пользователю переформировать столбцы микрокода для программирования любого ППЗУ;

AMMAP/29 — ассемблер символических данных ППЗУ устройства распределения. При использовании совместно с *AMDASM* программа *AMMAP* генерирует адреса точек входов в микропрограммы, которые загружаются в ОЗУ блока управления ЭВМ для дешифрации команд и генерации начальных адресов микропрограмм;

вспомогательные программы для микропрограммируемой системы.

После того как микропрограмма написана, проассемблирована и полученный микрокод загружен в оперативную управляющую память Системы 29, пользователю необходимо установить точки

переходов, биты прерываний, сравниваемые величины и т. д. с тем, чтобы он мог проверить и отладить данный микрокод и связанные с ним аппаратные средства. *AMDOS*/29 содержит набор программ, позволяющих устанавливать прерывания, а также обеспечивать возможность «просматривать и перетасовывать» содержимое памяти микропрограмм. Примерами таких директив являются следующие:

Display (Воспроизвести) — отобразить содержимое памяти на дисплее оператора;

Move (Переслать) — пересылка блока данных из одного места памяти в другое;

Locate (Разместить) — размещение конкретной последовательности символов в памяти;

Store (Запомнить) — запись шестнадцатиричных данных в память;

Verify (Проверить) — проверка сравнением двух блоков данных в памяти;

Set (Установить) — установка описателей таблицы форматов микропрограмм;

Jump (Перейти) — передача управления любой ячейке оперативной управляющей памяти;

Read (Читать) — считывание данных шестнадцатиричного формата с перфоленты;

Write (Записать) — вывод данных шестнадцатиричного формата на перфоленту;

Page (Страница) — выбор конкретной страницы в качестве старших 32 Кбайтов пространства адресов МП *Am*9080А.

Директивы *Воспроизвести*, *Переслать*, *Разместить*, *Запомнить*, *Читать* и *Записать* выполняются для любого типа имеющейся памяти, т. е. памяти *Am*9080А или вспомогательных страниц в микропрограммируемой системе (памяти микропрограмм, памяти распределения контрольных страниц и т. д.).

Использование Системы 29. После завершения разработки основной логической системы разработчик должен определить формат микрокоманд и с помощью диалогового редактора создать исходные файлы микропрограмм. Окончив создание исходного файла, с помощью ассемблера разработчик может оттранслировать (ассемблировать) микропрограмму. В результате ассемблирования получается двоичное представление микропрограммы. Выходные данные ассемблера могут быть получены в нескольких различных форматах. Двоичный файл записывается на НГМД для дальнейшей обработки.

Двоичная микропрограмма (микрокод) может быть загружена в оперативную управляющую память для отладки разрабатываемого устройства. Когда отладка микропрограмм успешно завершается, с помощью программы *AMPROM*/29 можно вывести микропрограмму на перфоленту для программатора ППЗУ. Вывод можно произвести также непосредственно на программатор ППЗУ.

Система АРМ2-05. Комплекс АРМ2-05 является аппаратно-программным комплексом, поддерживающим процессы программирования и микропрограммирования. С его помощью можно осуществлять интегрированную отладку микропрограмм (программ) и схемного оборудования в составе различных цифровых устройств (прежде всего построенных на базе микропроцессорных комплектов) [26].

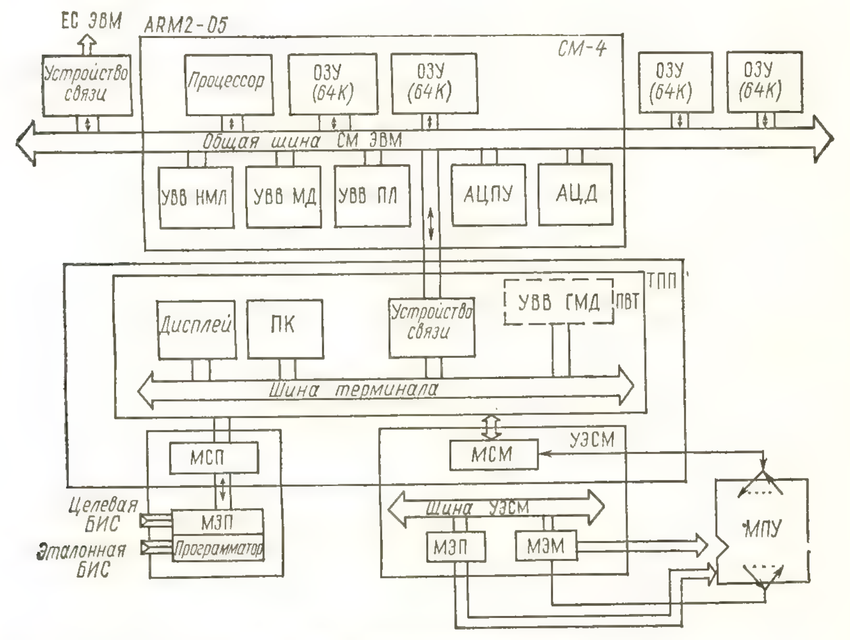


Рис. 10.10. Структурная схема комплекса АРМ2-05

Общая структура комплекса представлена на рис. 10.10. Обозначения на рисунке: *МСМ* — модуль связи с макетом; *МСП* — модуль связи с памятью; *МЗП* — модуль записи в память; *УЭСМ* — устройство эмуляции и связи с макетом; *МЭП* — модуль эмуляции памяти; *МЭМ* — модуль эмуляции микропроцессора; *МПУ* — микропроцессорное устройство.

К центральной машине СМ-4 подключено до четырех рабочих мест — терминалов подготовки программ ТТП. Терминалы содержат микроЭВМ СМ1800 с подключенными специальными техническими средствами — *УЭСМ* и *программатором*. Базовое программное обеспечение (БПО АРМ2-05) распределено между двумя машинами (СМ-4 и СМ 1800) и обеспечивает автоматизацию таких процедур микропрограммирования (программирования), как написание микропрограмм, их ввод в систему, трансляция, отладка и документирование.

Наряду с целевыми функциями БПО АРМ2-05 включает мощные средства настройки параметрического и процедурного типов с соответствующими метаязыками.

В рамках создания первой очереди комплексов разработаны и серийно освоены два аппаратно-программных комплекса прикладных модулей (КПМ), ориентирующих АРМ2-05 на работу с микропроцессорными сериями К580 (КПМ-80) и К589 (КПМ-89). Комплект КПМ-89 может применяться для эмуляции микропрограммной управляющей памяти в *МПУ*, использующих и другие секционированные микропроцессоры (отличные от К589).

Интеллектуальные возможности *ТПП* сделали реальным переход от его только комплексного использования в составе групповых АРМ2-05 к возможности автономного использования в качестве персонального мини-АРМ. Для этих целей *ТПП* (персональный АРМ2-05) оснащают дополнительным комплектом БПО с функциями построителя трансляторов (прежде всего ассемблеров, микроассемблеров на базе языка Микросленг и трансляторов с проблемно-ориентированных языков), а также с расширенными функциями подготовки кодов для занесения в ППЗУ.

Включение в ряд АРМ2-05 наряду с групповыми комплексами персональных комплексов является основным элементом работ по созданию второй очереди АРМ2-05. На этом же этапе разрабатывается комплект КПМ-04 для поддержки МПК серии К1804, а также осуществляется переход на новую модификацию программатора, позволяющего программировать практически всю серию КР556 (в первой очереди можно было работать только с КР556РТ4).

Возможности КПМ-04 выходят за область проектирования на базе МПК БИС серии К1804. Более широкое применение КПМ-04 обусловливается соответствующим программным обеспечением (в том числе мета-микроассемблером), и особенностями аппаратных модулей связи с макетом *УЭСМ*. Эти модули программным путем могут переключаться из режима эмуляции микросхем выбора следующего адреса, входящих в комплект К1804, в так называемый режим «универсального стенда». Этот режим обеспечивает возможность подключения к АРМ2-05 *МПУ*, построенных на произвольной микропрограммной базе. Для этого достаточно, чтобы сигналы шины адреса микрокоманды и самой микрокоманды в *МПУ* были доступны извне.

Возможность автономного применения персонального АРМ2-05 вовсе не исключает его работы в составе САПР с распределенной обработкой информации. Решение этой задачи — одно из направлений третьей очереди развития АРМ2-05.

Длина микрокоманды — до 128 разрядов.

На основе различных вариантов АРМ2-05 можно создавать различные проблемно-ориентированные технологии микропрограммирования. Однако все эти технологии в той или иной степени должны включать следующие этапы.

1. Настройка. Разработка специальных проблемно-ориентированных пакетов программ и аппаратных модулей; настройка языка микропрограммирования Микросленг или введение в систему новых ЯМП. Микросленг — настраиваемый язык среднего уровня, позволяющий свободно оперировать при написании программ (микропрограмм) возможностями языков низкого и структурированного высокого уровней.

Кроме этого, настройка лингвистического обеспечения включает:

описание на языке РТРАН (или на традиционном языке программирования, если возможностей РТРАН недостаточно) синтаксиса и некоторых элементов семантики специальных языков (например, таким образом можно расширить множество отладочных директив);

описание на языке Микропроект системы микрокоманд, связи конструкций Микросленг со структурой микрокода;

описание на языке Формат форматов выходных документов.

2. Кодирование. Кодирование осуществляется на языках, определенных в фазе настройки.

3. Автономная отладка. Под этим термином понимается отладка микропрограмм с помощью программной модели устройств. Она может осуществляться на двух основных уровнях: системного промежуточного языка (F-языка) и на уровне двоичных кодов. Для моделирования на уровне кодов требуется разработка специального интерпретатора в каждом конкретном случае.

4. Комплексная отладка (КА). Под этим термином понимается отладка микропрограмм с учетом реальной среды и в реальном масштабе времени на макете реального устройства, связь с которым осуществляется с помощью УЭСМ.

При КА микрокоды загружаются в имитатор памяти и инициируется процесс исполнения микропрограммы в различных режимах. Управление отладкой в том или ином режиме осуществляется с помощью языка общения с системой Микрод. Характерной особенностью отладки является доступ к целевой программе (микропрограмме) на уровне средств исходного языка Микросленг. Это касается как внесения изменений, так и задания различных режимов отладки. Эмуляция микропроцессора осуществляется заменой целевого микропроцессора в макете инструментальным микропроцессором того же типа, помещенным в специальный блок УЭСМ. Подключение инструментального микропроцессора к схемам макета производится кабелем, вилка которого вставляется в микропроцессорное гнездо на печатной плате макета. Эмуляция памяти осуществляется заменой всей или определенной области целевой памяти инструментальной памятью, выполненной на модулях ОЗУ. Блок эмуляции памяти включает также средства подключения инструментальной памяти к внутренней шине целевого устройства, сохраняя при этом возможность обращения к ней и по внутренней шине УЭСМ.

Директивы отладки предоставляют пользователю следующие возможности;

доступ по чтению-записи информации к переменным, массивам, регистрам, ячейкам памяти и аппаратно доступным точкам макета;

выдача трассы;

зацикливание программы по времени и по контрольным точкам;

задание остановов (по чтению или записи переменной или регистра, по *n*-му прохождению точки, по ветвлению, по маске и т. д.);

локальное внесение изменений без перетрансляции исходных данных.

Можно предположить, что директивы, реально доступные в каждом конкретном случае, могут зависеть от типа УЭСМ и что использованию тех или иных директив должен предшествовать процесс настройки.

5. Выпуск документации (в том числе и технологической для изготовления сменных слоев маскируемых ПЗУ). Возможно непосредственное изготовление БИС памяти с помощью программатора.

Система ориентирована на работу в диалоговом режиме.

Система МЕТАМИКРО. В основу организации системы положены следующие основные концепции [28].

1. Модульность. Отдельные программные и аппаратные компоненты системы представляют собой модульные изделия, решающие функционально законченные задачи. Программные модули выполнены совместимыми по форматам данных и алгоритмам обмена, аппаратные имеют стандартизованные интерфейсы. Совместимость модулей позволяет комплексировать их, формируя инструментальные комплексы различных конфигураций. Выбор приемлемой конфигурации осуществляет пользователь системы МЕТАМИКРО.

2. Поддержка процесса разработки микропрограмм и отладки аппаратуры. Система МЕТАМИКРО использует в своей работе метод внутрисхемной эмуляции. Важнейшее достоинство этого метода состоит в том, что он позволяет проводить отладку и проверку микропрограмм и аппаратуры разрабатываемого микропроцессорного устройства в тех же временных, климатических, вибрационных и прочих режимах, при которых оно будет впоследствии эксплуатироваться.

3. Настраиваемость. Система МЕТАМИКРО настраивается на архитектуру микропроцессорного устройства и символический язык его микропрограммирования пользователем с помощью специальных диалоговых процедур.

4. Независимость программных модулей от типа внутрисхемного эмулятора. Реализация этого принципа позволяет сократить число программных модулей в системе, что снижает трудоемкость

ее разработки. Достигается путем стандартизации системы команд внутрисхемного эмулятора.

5. Расширяемость. В связи со стандартизацией форматов данных и интерфейсов система МЕТАМИКРО открыта для расширения, обеспечивает простоту введения в ее состав новых модулей.

6. Простота эксплуатации. В целях воплощения этого принципа все компоненты системы, с которыми пользователь взаимодействует непосредственно, реализованы в наиболее удобном виде.

В первую очередь это касается метаязыка, который содержит все пять операторов, позволяя при этом описывать самые разнообразные форматы микрокоманд. Часть программных компонент системы работает с пользователем в режиме диалога, контролируя его действия. Ошибки, возникающие при описании символического языка микропрограммирования, написании микропрограмм и проведении отладки, обнаруживаются соответствующими модулями, которые выдают диагностические сообщения.

В общем случае работа с инструментальным комплексом, собранным из модулей МЕТАМИКРО, состоит в реализации следующих четырех фаз: настройки на символический язык микропрограммирования и архитектуру устройства; написания и трансляции микропрограмм; отладки микропрограмм и аппаратуры; документирования микропрограмм. Некоторые конфигурации инструментальных комплексов могут поддерживать не все указанные фазы или обеспечивать реализацию некоторых из них не в полных объемах.

Первая фаза начинается с разработки символического языка микропрограммирования. Описание символического языка вводится в программу-генератор, которая настраивает транслятор на работу с этим языком. Настройка осуществляется за одно исполнение программы-генератора.

Сгенерированный транслятор используется на фазе трансляции для перевода символических микрокоманд в машинную форму. Микрокоманды в машинной форме размещаются в ОЗУ эмулятора и исполняются в различных режимах, обеспечивающих эффективную отладку микропрограмм и аппаратуры. Отлаженные микропрограммы документируются и для них выводятся перфоленты, предназначенные для управления работой программатора.

Система МЕТАМИКРО оснащена настраиваемыми строчным ассемблером и кросс-ассемблером, что дает возможность пользователю организовать фазу трансляции в наиболее приемлемой для него форме. Она позволяет компоновать одноуровневую и двухуровневую конфигурации. Одноуровневая конфигурация (рис. 10.11) имеет в качестве ядра микроЭВМ с такой же системой команд (например, СМ3, СМ4, ЭлектроникаНЦ-80). К ЭВМ подключают алфавитно-цифровой дисплей АЦД устройства ввода-вывода перфоленты *УВВ ПЛ*, печатающее устройство, *внутрисхемный эмулятор* и программатор ППЗУ. Двухуровневая кон-

фигурация в дополнение к техническим и программным средствам одноуровневой содержит программные модули, исполняемые в ЭВМ ЕС. Связь между уровнями осуществляется с помощью перфоленты.

Описание символического языка микропрограммирования вводится пользователем в режиме диалога с *АЦД*. Оно воспринимается программой-генератором строчного ассемблера и дизассемблера, которая настраивает строчный ассемблер и дизассемблер на работу с символическими микрокомандами. Далее пользова-

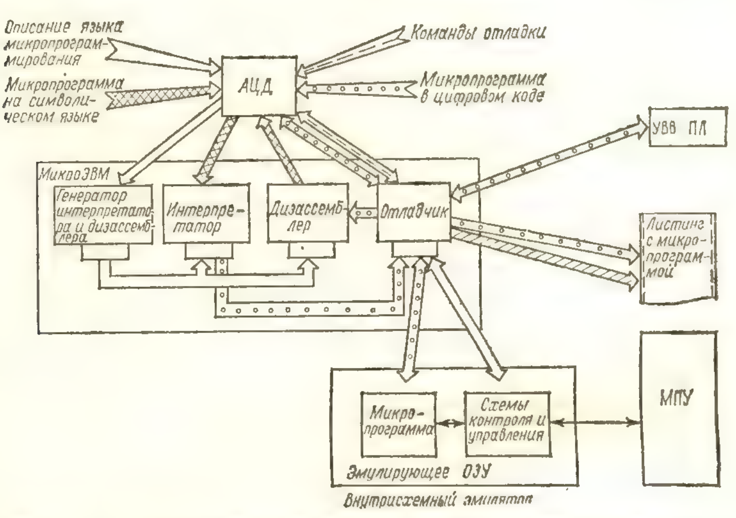


Рис. 10.11. Схема технологического процесса в одноуровневой системе МЕТАМИКРО

тель набирает на клавиатуре дисплея символическую микропрограмму, которую строчный ассемблер вначале транслирует в машинный код, а затем отладчик размещает в *эмулирующем ОЗУ*. Трансляция каждой микрокоманды осуществляется независимо, поэтому все ее поля, включая и адресные, должны быть полностью определены. Это вынуждает пользователя заботиться о присвоении адресов каждой микрокоманде. Протранслированная микрокоманда может быть выведена на экран дисплея в символическом виде с помощью дизассемблера, исполнена и, если необходимо, изменена и повторно протранслирована. При необходимости оттранслированная микропрограмма выводится на перфоленту в двоичном коде.

Отладка микропрограммы и аппаратуры осуществляется с помощью отладчика, который через монитор воспринимает вводимые пользователем директивы отладки и взаимодействует с внутрисхемным эмулятором. Директивы отладчика позволяют читать любую ячейку эмулирующей памяти в двоичной и восьмеричной

системах счисления, а также выводить их содержимое на экран в символическом виде с помощью дизассемблера. Отладчик дает возможность читать содержимое регистров общего назначения, слово состояния микропроцессора, распечатывать любую область эмулирующей памяти, измерять времена исполнения различных ветвей микропрограммы с помощью таймера, просматривать уровни сигналов в различных точках отлаживаемого устройства, исполнять микропрограмму в покомандном режиме и с остановами по заданным адресам либо событиям, а также исполнять ряд других

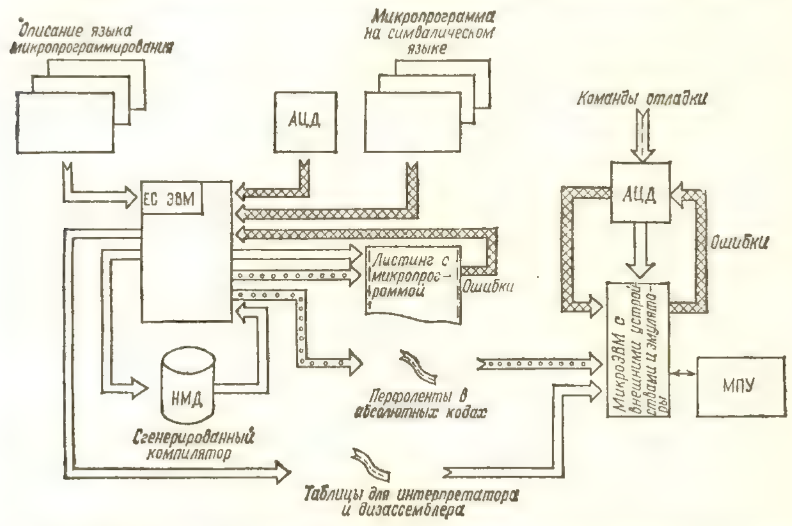


Рис. 10.12. Схема технологического процесса в двухуровневой системе МЕТАМИКРО

действий, позволяющих эффективно отлаживать микропрограммы и аппаратуру. Отлаженная микропрограмма выводится на перфоленту в виде, необходимом для работы программатора, а ее текст и двоичные коды документируются на печатающем устройстве.

Основным достоинством технологического процесса, обеспечиваемого одноуровневым инструментальным комплексом, является то, что он охватывает полный цикл разработки микропрограмм и отладки аппаратуры на индивидуальном рабочем месте, причем взаимодействие с комплексом осуществляется на микроассемблере, что значительно повышает производительность труда (в его составе отсутствует НГМД). Вместе с тем ограниченные ресурсы памяти микроЭВМ не позволяют разместить в ней эффективный транслятор перемещающего типа.

Технологический процесс в двухуровневом инструментальном комплексе происходит следующим образом (рис. 10.12). Символический язык микропрограммирования описывается по формальным

правилам системы МЕТАМИКРО на метаязыке и вводится в программу-генератор кросс-ассемблеров, размещенную в ЕС ЭВМ. Результатами работы генератора кросс-ассемблеров являются сгенерированный кросс-ассемблер и специальные таблицы для строчного ассемблера и дизассемблера, настраивающие их на символику языка. Кросс-ассемблер размещают на диске в ЕС ЭВМ, а таблицы, выданные генератором на перфоленту, вводятся в микроЭВМ.

Исходную микропрограмму вводят в ЕС ЭВМ двумя способами. Во-первых, микропрограмма может быть записана на стандартном бланке программирования, затем отперфорирована на перфокартах, которые далее и обрабатываются кросс-ассемблером, во-вторых она может быть набрана на клавиатуре АЦД обслуживаемого стандартной диалоговой системой ЕС ЭВМ (например, Примус, СРВ).

Результаты трансляции выводят на АЦПУ и содержат текст микропрограммы на символическом языке микропрограммирования и машинные коды микрокоманд. При наличии в оттранслированной микропрограмме синтаксических ошибок их исправляют и осуществляют повторную трансляцию. После устранения всех ошибок выводят перфоленту с абсолютным кодом микропрограммы, которую загружают в микроЭВМ для отладки.

Двухуровневый инструментальный комплекс обладает бóльшими логическими возможностями в фазе трансляции микрокоманд, чем одноуровневый, что объясняется наличием в его составе мощного кросс-ассемблера, размещенного в ЕС ЭВМ. Однако наличие ЭВМ второго уровня в инструментальном комплексе снижает его оперативные возможности, так как доступ к ЕС ЭВМ всегда приходится разделять с другими пользователями машины.

Разработку микропрограмм с помощью системы МЕТАМИКРО начинают с разработки символического языка микропрограммирования. Система МЕТАМИКРО предоставляет широкие возможности в построении языка, практически не ограничивая пользователя в символике, количестве форматов микрокоманд, числе полей микрокоманды, в их длине и порядке следования.

Пользователь создает язык из следующих элементов.

1. Символическое имя. Составляется из одного — восьми любых символов кода ДКОИ, имеющихся на устройстве подготовки данных. Является символическим эквивалентом цифрового кода микроопераций. Допускается присвоение одному цифровому коду нескольких имен.

2. Символическая метка. Составляется по тем же правилам, что и символическое имя.

3. Число. Целое положительное число можно записать по выбору пользователя в двоичной, восьмеричной, десятичной или шестнадцатиричной системах счисления.

4. Символ *. Обозначение текущего значения счетчика микрокоманды.

5. Разделители. В качестве разделителей используются символы: ␣ (пробел) и / (наклонная черта) — разделитель между полями микрокоманды; : (двоеточие) — разделитель, записываемый после символической метки; ; (точка с запятой) — разделитель, записываемый в конце микрокоманды.

6. Знаки арифметических операций: + (плюс) и — (минус).

Допустимая в системе МЕТАМИКРО микрокоманда имеет формат [⟨метка 1⟩ :] ... [⟨метка n⟩ :] ⟨поле 1⟩ $\left\{ _ \atop / \right\}$ ... ⟨поле m⟩.

Микрокоманда состоит из полей, отделенных друг от друга пробелами ␣ или наклонной чертой /, и завершается символом ; .

Поле, представляющее собой часть микрокоманды, характеризуется типом и длиной. Длина поля есть число отведенных ему двоичных разрядов и не должна превышать 31 разряда, суммарная длина всех полей, т. е. длина микрокоманды, — не более 128 разрядов.

В системе МЕТАМИКРО допускаются поля следующих типов: *VAR* — поле переменных, значение которого определяется подставленным в него символическим именем; *DIG* — поле, значение которого задается числом (числовое поле); *SYM* — поле, значение которого означает адрес перехода, задается либо в виде символической метки, либо с использованием текущего значения счетчика микрокоманд. В последнем случае поле определяется одной из форм: *, * + число, * — число.

Разрешается использовать поля, составленные из любых комбинаций рассмотренных выше типов.

В поле типа *VAR* могут находиться символические имена K145, *SALUT*, *START*, в поле *DIG* допускаются числа вида #*B1∅1∅*, #*Q*17, #*D*49, #*HF*5. Поле типа *SYM* объявляется полем, которое может содержать адрес перехода в виде символической метки, например *M*22. Это поле может также содержать информацию вида *, * + #*B1∅*, * — #*D*25.

Совокупность отличительных особенностей, определяющих правила записи группы микрокоманд, отражается в формате микрокоманды, который определяется пользователем и требует для своего однозначного задания указания следующих параметров: длины микрокоманды; числа полей в микрокоманде; порядка следования полей; символического имени каждого поля.

Рассмотрим примеры микрокоманд, записанных с учетом требований системы МЕТАМИКРО: *M*1 : ␣ *M*22 : ␣ R2 = R4 ␣ + + ␣ R5␣; .

В данной микрокоманде в качестве разделителя полей использован пробел. Микрокоманда содержит четыре поля, в которых записаны символические имена: R2=, R4, + (плюс), R5. Микрокоманде предшествуют две метки *M*1, *M*22.

При использовании в качестве разделителя полей наклонной черты микрокоманда имеет вид: *M*1 : ⎵ *M*22 : ⎵ *R*2 = /*R*4/ + + /*R*5 ⎵; .

В следующей микрокоманде последнее поле содержит число, заданное в десятичной системе счисления: *AAA* = ⎵ *AAA* ⎵ — — *CONST* ⎵ *CONST* = ⎵ # *D*25 ⎵; .

Использование символа * для организации адреса перехода иллюстрируется такими командами: *IN* = ⎵ *OUT*1 ⎵ *SAVE* ⎵ ⎵ * + # *D*2 ⎵; *X* ⎵ *X* ⎵ *X* ⎵ *GOTO* ⎵ * — *HFF* ⎵; .

Последняя микрокоманда содержит пять полей, три из них, помеченных символом *X*, не используются.

Разработанный пользователем язык должен быть записан по формальным правилам для того, чтобы система могла автоматически сгенерировать транслятор. Совокупность этих формальных правил образует своего рода язык, который по отношению к языку пользователя является метаязыком.

Метаязык содержит следующие основные директивы:

1. Директива *LENGTH* задает длину в битах самой длинной микрокоманды языка. Формат: *LENGTH* ⎵ ⟨длина⟩. Например: *LENGTH* ⎵ 64.

2. Директива *FORMAT* задает структуру микрокоманды, т. е. описывает ее длину, число полей, порядок их следования и символические имена. Структура директивы: *FORMAT* ⎵ ⎵ ⟨*N*⟩ ⎵ ⟨*длина*⟩ ⎵ ⟨*имя поля* 1⟩, ..., ⟨*имя поля m*⟩; . *N* — номер формата; длина — длина микрокоманды в битах, выраженная в десятичной системе счисления, имя поля — символическое имя поля (идентификатор). Например: *FORMAT* ⎵ 1 ⎵ ⎵ 24 ⎵ A, B, *RESULT*, *OPER*, *CONST*, *BRANCH*; .

Если для нескольких форматов идентификаторы ряда полей совпадают, то повторно эти поля описывать не надо. Вместо любого повторяющегося идентификатора поля может стоять символ *, который означает, что вся информация, характеризующая поле, содержится в директиве *FIELD*, относящейся к предыдущей директиве *FORMAT*.

3. Директива *FIELD* задает длину и тип поля. Структура директивы: *FIELD* ⎵ ⟨*имя поля*⟩ ⎵ ⟨*длина поля*⟩ ⎵ ⟨*тип поля*⟩; .

Имя поля должно совпадать с одним из идентификаторов полей, заданных в директиве. Длина поля отражает число разрядов в нем и задается в десятичной системе счисления. Тип поля может быть одним из семи указанных ранее.

Например: *FIELD* ⎵ *A* ⎵ 4 ⎵ *VAR*; *FIELD* ⎵ *CONST* ⎵ ⎵ 16 ⎵ *DIG*, *SYM*, *VAR*; .

4. Директива описания поля задает символические имена микроопераций и соответствующие им цифровые значения, всегда должна следовать за директивой FIELD, в которой в характеристике поля присутствует тип VAR.

Структура директивы:
⟨имя⟩ : ⟨число⟩, ⟨имя⟩ : ⟨число⟩, ..., ⟨имя⟩ : ⟨число⟩;

Например: R1 : # B ∅ 1, R2 : # B ∅ 2; CA : # H1F, D : : # Q77; .

Исходные данные для генерации компилятора представляют совокупность описанных выше директив и образуют последовательный файл с длиной записи не более 80 символов. Кроме операторов метаязыка файл может содержать комментарии.

В процессе генерации кросс-ассемблера система МЕТАМИКРО контролирует вводимую информацию и выдает сообщения об ошибках.

После завершения разработки символического языка микропрограммирования и его описания на метаязыке генерируется кросс-ассемблер для этого языка. Генерация проводится в ЭВМ специальными программами-генераторами, которые формируют таблицы символов для кросс-ассемблера и некоторые его программы. С этого момента компилятор становится законченным самостоятельным программным продуктом.

Генерация строчных ассемблеров и дизассемблеров может быть осуществлена как в ЕС ЭВМ, так и непосредственно в ЭВМ первого уровня. В ЕС ЭВМ таблицы для строчного ассемблера и дизассемблера являются результатом работы генератора компиляторов, которые порождают их на основе описания символического языка. В ЭВМ первого уровня генерация строчных ассемблеров и дизассемблеров осуществляется специальным программным модулем, реализующим этот процесс в диалоге. Генератор начинает работу по директиве пользователя и может работать в режимах: ввод данных; корректировка данных; генерация таблиц; вывод таблиц на перфоленту и документирование.

В режиме ввода создаются все необходимые для работы строчного ассемблера и дизассемблера таблицы в так называемом исходном формате, который характеризуется неэкономным использованием памяти ЭВМ, однако обеспечивает удобство процессов ввода и корректировки данных.

Проверенные и откорректированные таблицы в исходном формате служат материалом для генерации. Процесс генерации таблиц представляет собой, по сути дела, их сжатие в целях приведения к специальному формату, с которым работают интерпретатор и дизассемблер. Таблицы в исходном и сжатом форматах могут быть выданы на перфоленту в виде, стандартном для ЭВМ, а также задокументированы. Процедура генерации начинается выдачей директивы *B*. Система запрашивает число форматов микрокоманд, а затем число полей для каждого формата, после чего начинается процесс занесения данных, относящихся к первому формату. Задается длина формата и для каждого поля его длина и тип. Для полей типа *VAR* вводят символические обозначения микроопераций и их цифровые коды. Таблицы в исходном формате могут быть выведены на перфоленту и распечатаны, что дает возможность прервать процесс занесения таблиц в желаемом месте и продолжить его с этого же места в следующем сеансе работы с инструмен-

тальным комплексом. В этом случае работа должна быть продолжена в режиме корректировки. Корректировке подлежат только таблицы в исходном формате. Процедура корректировки вызывается директивой *K* _ ⟨номер формата⟩ ⟨номер поля⟩.

Параметры директивы определяют номер корректируемого поля и номер формата, к которому это поле относится. После выдачи директивы генератор распечатывает номер строки таблицы символов и содержимое этой строки.

Пользователь имеет возможность ввести новое содержимое или просматривать таблицы последовательно по строкам.

Кросс-ассемблер выполняется на ЕС ЭВМ под управлением операционной системы ОС. В результате трансляции выдается листинг исходного текста, сопровождаемый диагностическими сообщениями об обнаруженных ошибках, а также листинг с двоичными кодами микропрограмм. Кроме того, может быть выдана перфолента с двоичными кодами микропрограммы.

Для управления процессом кросс-ассемблирования и организации данных в микропрограмму можно включать следующие директивы:

OUT — управление выводом при компиляции;

PAGE — формирование заголовка страницы листинга микропрограммы;

C — комментарии;

ORG — установка счетчика адресов микрокоманд;

FORMAT — установка формата для группы микрокода;

END — конец микропрограммы.

Микропрограмму (если ее не вводят с АЦД) записывают на стандартном бланке, имеющем 80 позиций в каждой строке. Для размещения информации используют позиции 1 ... 72. Позиции 73 ... 80 компилятором не рассматриваются и могут быть зарезервированы для идентификации строк. Расположение информации в строке не фиксировано какими-либо графами.

Между элементами языка можно помещать любое количество пробелов.

Каждый оператор микропрограммы должен начинаться с новой строки. Операторы-директивы занимают по одной строке, операторы-микрокоманды могут размещаться в нескольких строках, при этом элемент языка не разделяют между строками. Директивы размещают в строке так, чтобы их код начинался обязательно с первой позиции. В отличие от них микрокоманды должны иметь в первой позиции строки пробел. Группе микрокоманд одного формата должна предшествовать определяющая их директива.

Кросс-ассемблер подготавливают к использованию в форме отредактированного загрузочного модуля, занесенного в личную библиотеку. Входными данными для трансляции служит текст исходной микропрограммы на перфокартах или на магнитном диске. Кросс-ассемблер может отрабатывать несколько микропрограмм, разделенных директивой END. В этом случае для

каждой последующей микропрограммы, которая не содержит соответствующих директив, сохраняются параметры режима вывода и обозначение документа, установленные для предыдущей микропрограммы, а также продолжается нумерация листов.

Листинги микропрограммы выдаются на листах формата 12, оформленных в соответствии с требованиями к программным документам. Исходный текст содержит пронумерованные строки текста микропрограммы, которые сопровождаются диагностическими сообщениями компилятора об обнаруженных в результате синтаксического анализа ошибках. Микрокоманды снабжаются адресом, представленным в восьмеричной системе счисления. Протокол содержит также таблицу присвоения меткам адресов.

Машинный код микропрограммы распечатывается в двоичной системе счисления. Каждая микрокоманда предваряется адресом, поля внутри микрокоманды разделяются пробелом. Микропрограмма в машинных кодах выдается на перфоленту блоками, каждый из которых содержит массив микрокоманд, занимающий непрерывный участок памяти с заданного адреса.

Диагностические сообщения об обнаруженных кросс-ассемблером синтаксических ошибках выдаются сразу же за текстом соответствующей строки.

Строчное ассемблирование и дизассемблирование микропрограмм реализуется модулями, исполняемыми в ЭВМ первого уровня. Если дизассемблер в каком-либо поле микрокоманды обнаруживает цифровой код, для которого в таблицах символов отсутствует символическое обозначения, то этот код выдается на экран, обрамленный слева и справа вопросительными знаками. Если после выдачи директивы дизассемблирования на клавиатуре дисплея нажимается клавиша Возврат, то осуществляется дизассемблирование микрокоманды, расположенной по следующему адресу. Нажатие клавиши / после дизассемблирования инициирует строчный ассемблер, который размещает вводимую микрокоманду по адресу дизассемблированной.

Строчный ассемблер выполняет перевод в двоичную форму каждого поля микрокоманды в отдельности. Если символы поля отсутствуют в таблице строчного ассемблера, то на экран выводится знак ?. После этого можно повторно вводить данное поле. Если пользователь обнаруживает ошибку в поле до завершения его ввода, то он может откорректировать символы, сдвигая маркер в позицию, с которой надо ввести правильные данные. Содержимое поля типа *DIG* можно вводить в двоичной, восьмеричной и шестнадцатиричной системах счисления.

Отладка микропрограмм и аппаратуры осуществляется пользователем с помощью программы-отладчика, расположенной в микроЭВМ. Фаза отладки начинается с загрузки в эмулирующее ОЗУ микропрограмм или микропрограммных тестов.

Отладчик позволяет исполнять микропрограмму в четырех режимах: автоматическом с пуском с заданного адреса; автомати-

ческом с пуском с заданного адреса и остановом по событию; пошаговом, начиная с заданного адреса.

Для отладки аппаратуры эмулятор снабжен трассировочным ОЗУ, позволяющим контролировать состояния 32 точек отлаживаемого устройства в течение последних 16 тактов его функционирования. Содержимое трассировочного ОЗУ выдается на экран дисплея в двоичном коде. Единичные состояния ячеек трассировочного ОЗУ отражают высокие потенциалы на контролируемых точках, нулевые — низкие. Для отладки аппаратуры возможны многократное исполнение любой микрокоманды и организация прерываний в любой точке микропрограммы. Полное представление о возможностях отладчика дает рассмотрение его директив.

Директива ЧИТАТЬ И МОДИФИЦИРОВАТЬ МИКРОКОМАНДУ осуществляет чтение и запись данных в эмулирующее ОЗУ. Формат: *R* ⟨система счисления⟩/⟨адрес микрокоманды⟩.

Поле системы счисления определяет, в каком виде выдается содержимое ячейки на экран. Если в этом поле стоит символ 2, то микрокоманда выводится в двоичном виде, если символ 8, то в восьмеричном. Содержимое микрокоманды расположено сразу за текстом директивы. Например:

R2/001766/0111000 ... 110

R8/001766/340 ... 6.

После отработки директивы пользователь может либо просматривать следующую микрокоманду, нажав клавишу Возврат, либо перейти к началу отладчика, нажав клавишу *. Для изменения содержимого микрокоманды пользователь вводит символ /. После ввода очередной цифры адреса микрокоманды или любого ее поля на экран выдается порядковый номер этой цифры, тем самым уменьшая вероятность ошибки пользователя.

Директива ПУСК С ЗАДАННОГО АДРЕСА В АВТОМАТИЧЕСКОМ РЕЖИМЕ. Формат: *G* ⟨пусковой адрес⟩/.

Директива ОСТАНОВ ПО ЗАДАННОМУ АДРЕСУ. Формат: *F* ⟨адрес останова⟩/.

Если эта директива была выдана раньше директивы ПУСК С ЗАДАННОГО АДРЕСА В АВТОМАТИЧЕСКОМ РЕЖИМЕ, то после останова микропрограммы печатается адрес останова, например G001347/001360, где 001347 — адрес пуска; 001360 — адрес останова.

Директива ПУСК С ЗАДАННОГО АДРЕСА ПО ШАГАМ. Формат: *ST* ⟨пусковой адрес⟩/.

По этой директиве будет выполнена одна микрокоманда, расположенная по пусковому адресу.

Директива ПРОДОЛЖИТЬ. Формат: *C*. Вызывает выполнение одной микрокоманды, если раньше была введена директива *ST*... Директивы *ST*... и *C* позволяют просматривать выполнение микропрограммы по шагам. После обработки одной микрокоманды на экран выводится адрес следующей.

Директива ОТМЕНИТЬ ОСТАНОВ. Формат: *B*. Вызывает отмену останова, заданного ранее директивой *F*.

Директива ЧИТАТЬ ТРАССИРУЮЩЕЕ ОЗУ. Формат: *TR*. Вызывает распечатку содержимого трассирующего ОЗУ.

Директива ЧИТАТЬ РЕГИСТРЫ ОБЩЕГО НАЗНАЧЕНИЯ. Формат: *RON*. Для различных типов микропроцессорных наборов форматы печати отличаются.

Директива ЧИТАТЬ СТАТУСНЫЙ РЕГИСТР. Формат: *SW*. В ответ выдается содержимое регистра состояния микропроцессора в виде PSW = ...

Директива ЧИТАТЬ ТАЙМЕР. Формат: *TM*. Выдается число микрокоманд, исполненных от пуска до останова микропрограммы.

Директива ВВЕСТИ ПЕРФОЛЕНТУ. Формат: *IN*. Перфолента вводится в эмулирующее ОЗУ. В процессе ввода перфолента контролируется и при обнаружении ошибок на экран выдаются сообщения.

Директива ВЫВЕСТИ ПЕРФОЛЕНТУ. Формат: *W* ⟨начальный адрес⟩/⟨конечный адрес⟩. Выводится на перфоленту содержимое эмулирующего ОЗУ от начального до конечного адресов.

Директива ЧИТАТЬ ЭМУЛИРУЮЩЕЕ ОЗУ. Формат: D ⟨система счисления⟩/⟨начальный адрес⟩/⟨конечный адрес⟩/.

Директива МНОГОКРАТНО ВЫПОЛНИТЬ МИКРОКОМАНДУ. Формат: E ⟨адрес микрокоманды⟩.

Директива ОСТАНОВ ПО СОБЫТИЮ. Формат: *Z* ⟨3 — разрядное двоичное число⟩/.

Число, указанное в директиве, заносится в компаратор, расположенный в эмуляторе. Три выхода компаратора подключают к трем точкам отлаживаемой схемы. Микропрограмму запускают директивой G... Останов микропрограммы произойдет в том случае, если потенциал на проводах образует цифровой код, который задан в директиве.

Директива ОТМЕНИТЬ ДИРЕКТИВУ представляет собой символ . . Выдается в момент, когда оператор пожелает прекратить набор какой-либо директивы. Вызывает передачу управления монитору.

Система МИКРАС [5, 6]. Основной отладочный инструмент — программная модель аппаратуры. Моделируемая схема представляется в виде совокупности базовых элементов, связанных между собой системой соединений. В качестве базовых элементов приняты типовые ИС и БИС. Модели базовых элементов, из которых собирают моделируемые устройства, организованы в виде библиотеки подпрограмм, реализующих алгоритм их функционирования. В состав библиотеки могут быть включены любые алгоритмически описанные функциональные узлы или схемы.

СЛОВАРЬ ТЕРМИНОВ

Адресное запоминающее устройство — ЗУ, в котором информация считывается по физическому адресу, т. е. по месту ее хранения.

Аккумулятор (А) — регистр АУ или АЛУ, в котором при выполнении какой-либо операции с двумя операндами находится один из операндов, а после выполнения — результат.

Аналоговая интегральная микросхема — ИС, предназначенная для преобразования и обработки сигналов по закону непрерывной функции.

Арифметико-логическое устройство (АЛУ) — функциональная часть процессора, выполняющая арифметические и логические действия над данными, и реализованная в виде одной БИС или ее части.

Арифметический расширитель (АР) — функциональная часть процессора, которая совместно с АУ или АЛУ обеспечивает возможность аппаратного или микропрограммного выполнения сложных арифметических операций и может быть реализована в виде одной БИС или ее части.

Арифметическое устройство (АУ) — функциональная часть процессора, выполняющая арифметические действия над данными и реализованная на одной БИС или ее части.

Архитектура вычислительной системы — общая логическая организация ЦВС, определяющая процесс обработки данных в конкретной ВС и включающая методы кодирования данных, состав, назначение, принципы взаимодействия технических средств и программного обеспечения.

Ассоциативное запоминающее устройство (АЗУ)— ЗУ, поиск информации в котором производится не по ее физическому адресу, а по содержимому.

Базовый комплект БИС — минимальный состав комплекта, необходимый для построения основных узлов радиоэлектронной аппаратуры.

Базовый МПК — минимальный состав МПК, необходимый для построения основных узлов микропроцессора или контроллера.

Бит — элемент кодового набора, состоящего из двух элементов (0 и 1).

Блок микропрограммного управления (БМУ) — часть микропрограммно-управляемого процессора, осуществляющая автоматическое управление процессом обработки данных в соответствии с программой.

Большая интегральная микросхема (БИС) — ИС, содержащая 500 и более элементов, изготовленных по биполярной технологии, 1000 и более элементов, изготовленных по МДП-технологии.

Время обращения к ЗУ — интервал времени между последовательными обращениями.

Данные — информация, представленная в виде, пригодном для обработки автоматическими средствами при возможном участии человека.

Динамическое запоминающее устройство — ЗУ, в котором для сохранения информации необходима периодическая подача сигналов управления.

Емкость ЗУ — наибольшее количество информации, которое одновременно может храниться в ЗУ. Измеряется в битах, байтах или машинных словах.

Заказная БИС — БИС, которая может быть применена при построении только одного изделия либо узкого класса средств обработки информации.

Запоминающее устройство (ЗУ) — устройство, реализующее функцию памяти данных.

Запоминающее устройство магазинного типа (магазинное ЗУ) — ЗУ, состоящее из группы связанных между собой ячеек, схематически расположенных

в колонку, в которой только верхняя имеет связь с остальной системой. При передаче данных в ЗУ или из него его содержимое передвигается вниз или вверх по колонке, освобождая или заполняя ячейки.

Интегральная микросхема (ИМС) — микроэлектронное изделие, выполняющее определенную функцию преобразования и обработки сигналов и (или) накапливания информации и имеющее высокую плотность упаковки электрически соединенных элементов (или элементов и компонентов) и (или) кристаллов, которое с точки зрения требований к испытаниям, приемке, поставке и эксплуатации рассматривается как единое целое. Синонимом термина интегральная микросхема является термин интегральная схема, или микросхема. Наиболее часто используют термин интегральная схема (ИС).

Интерфейс — совокупность средств и правил, обеспечивающих взаимодействие устройств ВС и (или) программ.

ИС первой степени интеграции — микросхема, содержащая до 10 элементов и компонентов включительно, *второй* — свыше 10 до 100 элементов и компонентов включительно, *третьей* — свыше 100 до 1000, *четвертой* — свыше 1000 до 10 000, *пятой* — свыше 10 000 до 100 000, *шестой* — содержащая свыше 100 000 до 1 000 000 элементов и компонентов включительно.

Кодовый набор — конечное множество, из элементов которого строят данные при кодировании.

Команда — указание, определяющее один шаг в общем процессе выполнения программы.

Комплект БИС — совокупность типов БИС, выполняющих различные функции, совместимых по архитектуре, конструктивному исполнению и электрическим параметрам и обеспечивающих их совместное применение при построении радиоэлектронной аппаратуры.

Компонент интегральной микросхемы — часть ИС, реализующая функцию электрорадиоэлемента, которая может быть выделена как самостоятельное изделие с точки зрения требований к испытаниям, приемке, поставке и эксплуатации.

Контроллер — часть периферийного или автономного устройства, представляющая цифровой автомат, который работает по определенному алгоритму, выполняет функции управления этим устройством по однозначно заданным алгоритмам в соответствии с ограниченным количеством поступающих извне команд, сопряженный с другими устройствами системы и не требует участия человека-оператора.

Корпус интегральной микросхемы — часть конструкции предназначенная для защиты микросхемы от внешних воздействий и для соединения с внешними электрическими цепями посредством выводов.

Логический элемент интегральной микросхемы — группы электрически соединенных элементов и (или) компонентов, реализующих одну из простейших функций алгебры логики.

Малая интегральная микросхема (МИС) — ИС первой и второй степени интеграции, содержащая один или несколько логических или аналоговых элементов (вентилей, триггеров, усилителей и т. д.).

Микрокалькулятор — миниатюрное, построенное на БИС вычислительное устройство с заданным при изготовлении определенным набором команд, в котором как команды, так и данные вводятся оператором с клавиатуры в определенном порядке.

Микрокоманда — совокупность совместимых микроопераций, выполняемых в жестко закоммутированной последовательности.

Микроконтроллер — контроллер, выполненный на микропроцессорах.

Микрооперация — элементарная операция, соответствующая элементарному машинному действию, обозначенному во внутреннем языке ЭВМ и не содержащая в себе других элементарных операций, обозначенных в этом языке.

Микропрограмма — последовательность микрокоманд, реализующих алгоритм выполнения команды.

Микропрограммное управление — способ организации работы устройства управления ЭВМ, при котором каждая команда при вводе ее в регистр команд определяет выполнение последовательности микрокоманд, в совокупности реализующих алгоритм выполнения команды.

Микропроцессор (МП) — программно-управляемое устройство, непосредственно осуществляющее процесс обработки цифровой информации и управление им, построенное на одной или нескольких БИС.

Микропроцессорная интегральная схема — ИС, выполняющая функцию микропроцессора или его части.

Микропроцессорная система — система, выполненная на микропроцессорах.

Микропроцессорная электронная вычислительная машина (микроЭВМ) — ЭВМ, состоящая из микропроцессора, полупроводниковой памяти, средств связи с периферийными устройствами, при необходимости, пульта управления и источника питания, объединенных общей несущей конструкцией.

Микропроцессорный комплект (МПК) — совокупность микропроцессорных и других ИС, совместимых по архитектуре, конструктивному исполнению и электрическим параметрам и обеспечивающих возможность совместного применения.

Обращение к ЗУ — операция, в результате которой происходит считывание или запись информации в ЗУ.

Оперативное запоминающее устройство (ОЗУ)— ЗУ, предназначенное для хранения информации, непосредственно участвующей в процессе выполнения программы и используемой процессором.

Операция — действия, в совокупности составляющие выполнение команды устройства ЭВМ или его реакцию на определенные условия.

Память данных — функциональная часть цифровой вычислительной системы (ЭВМ), предназначенная для приема, хранения и выдачи данных.

Полупроводниковая интегральная микросхема — ИС, все элементы и межэлементные соединения которой выполнены в объеме и на поверхности полупроводника.

Полупроводниковое запоминающее устройство — ЗУ, в котором для хранения информации используются полупроводниковые приборы. Реализуется в виде СИС, БИС и СБИС.

Пользователь ЦВМ — юридическое или фактическое лицо, применяющее ЦВМ.

Постоянное запоминающее устройство (ПЗУ) — ЗУ с неизменным содержимым памяти.

Программируемое постоянное запоминающее устройство (ППЗУ) — ПЗУ, в которое информация, подлежащая хранению, заносится однократно, после завершения процесса его изготовления.

Процессор — устройство или функциональная часть ЦВС, предназначенная для интерпретации программы.

Регистр (Рг) — функциональный узел, предназначенный для приема, хранения и выдачи коротких последовательностей двоичных знаков, объединенных общим признаком.

Регистр общего назначения (РОН) — программно-доступный для пользователя регистр процессора, который может быть использован при выполнении различных операций.

Сверхбольшая интегральная схема (СБИС) — ИС пятой и более высокой степени интеграции, содержащая одно или несколько функционально законченных устройств, выполняющих весь процесс обработки информации: хранение, преобразование, передачу и т. д. (например, микроЭВМ).

Сверхоперативное ЗУ (СОЗУ) — ОЗУ небольшой емкости, время обращения к которому незначительно по сравнению с временем операции.

Семейство комплектов БИС — совокупность комплектов БИС, обеспечивающих возможность построения архитектурно-совместимых устройств обработки данных.

Серия интегральных микросхем — совокупность типов ИС, которые могут выполнять различные функции, имеют единое конструктивно-технологическое исполнение и предназначены для совместного применения.

Система обработки данных — система, выполняющая автоматизированную обработку данных и включающая технические средства обработки данных методы и процедуры, программное обеспечение и соответствующий персонал.

Совместимые микрооперации — микрооперации, которые можно выполнять одновременно без искажения информации.

Средняя интегральная микросхема (СИС) — ИС второй и третьей степени интеграции, содержащая один или несколько функциональных узлов (регистров, счетчиков и т. д.).

Статическое запоминающее устройство — ЗУ, в котором информация сохраняется при отсутствии сигналов управления.

Степень интеграции микросхемы — показатель степени сложности микросхемы, характеризуемый числом содержащихся в ней элементов и компонентов. Определяется по формуле $K = \lg N$, где K — коэффициент, определяющий степень интеграции, значение которого округляется до ближайшего большего целого числа; N — число компонентов и элементов интегральной микросхемы, в том числе содержащихся в составе компонентов, входящих в интегральную микросхему.

Технические средства обработки данных — все оборудование, включая носители данных, предназначенное для автоматизированной обработки данных.

Унифицированная БИС — БИС, которая может быть применена при построении различных средств обработки информации.

Унифицированный комплект БИС — комплект БИС, который может быть применен при построении изделий электронной аппаратуры различных классов.

Управляющая электронная вычислительная машина (УВМ) — ЭВМ, которая, используя данные (информацию) о процессе, вырабатывает выходные сигналы, управляющие работой объектов, участвующих в этом процессе.

Центральная часть вычислительной системы (ВС) — часть средств ЦВС, в состав которой входят объединенные единым управлением ЦП, ОЗУ, каналы ввода-вывода, необходимые для выполнения программы после ее ввода.

Центральный процессор (ЦП) — процессор, выполняющий в данной ЦВС основные функции по обработке данных и управлению работой других частей этой системы.

Центральный процессорный элемент (ЦПЭ) — микропроцессорная БИС, непосредственно осуществляющая процесс обработки данных.

Цифровая вычислительная машина (ЦВМ) — часть ЦВС, представляющая ее технические средства, имеющая в своем составе одну центральную часть и предназначенная для обработки данных под управлением программы, находящейся в памяти.

Цифровая вычислительная система (ЦВС) — система обработки данных, настроенная на решение задач конкретной области применения и работающая с данными, представленными в кодированной форме.

Цифровая интегральная микросхема — ИС, предназначенная для преобразования и обработки сигналов, изменяющихся по закону дискретной функции.

Цифровой вычислительный комплекс (ЦВК) — часть ЦВС, представляющая ее технические средства, имеющая в своем составе более одной центральной части и предназначенная для обработки данных под управлением программы, находящейся в памяти.

Электрически программируемое постоянное запоминающее устройство (ЭППЗУ) — ПЗУ, в котором информация изменяется в результате воздействия электрических сигналов.

Элемент интегральной микросхемы — часть ИС, реализующая функцию какого-либо электрорадиоэлемента, которая выполнена нераздельно от кристалла или подложки и не может быть выделена как самостоятельное изделие с точки зрения требований к испытаниям, приемке и эксплуатации.

Энергонезависимое запоминающее устройство — ЗУ, в котором при отключении источников электропитания не происходит разрушение хранимой информации.

СПИСОК ЛИТЕРАТУРЫ

1. **Алексенко А. Г., Галицын А. А., Иванников А. Д.** Проектирование радиоэлектронной аппаратуры на микропроцессорах. М.: Радио и связь, 1984. 272 с.

2. **Аналоговые** и цифровые интегральные микросхемы. Справочное пособие/ С. В. Якубовский, И. А. Баринов, Л. И. Ниссельсон и др. Под ред. С. В. Якубовского. 2-е изд., перераб. и доп. М.: Радио и связь, 1985. 432 с.

3. **Архитектура** многопроцессорных вычислительных систем/Под ред. В. И. Тимохина. Л.: ЛГУ, 1981. 103 с.

4. **Балашов Е. П., Пузанков Д. В.** Микропроцессоры и микропроцессорные системы: Учеб. пособие для ВУЗов/Под ред. В. Б. Смолова. М: Радио и связь, 1981. 328 с.

5. **Бекасов А. А.** и др. МИКРАС — система микропрограммирования микроЭВМ, построенных на базе серии К589//Алгоритмы: Ташкент, 1980. С. 60—62

6. **Бекасов А. А.** и др. О развитии систем отладки микропроцессоров//Машинное моделирование. М.: МДНТП им. Ф. Э. Дзержинского, 1980. С. 31—33.

7. **Беднов В. Г., Пономарев Е. И., Симхес В. Я.** Оценка амплитуды перекрестных помех в межэлементных линиях связи ЭВМ//Вопросы радиотехники. Сер. ЭВТ, 1978, вып. 12. С. 131—137.

8. **Белов В., Левитский В.** ЭВМ — на суда//Морской флот, 1983, № 4. С. 22—24.

9. **Бобков В. А.** и др. Микромощные микропроцессорные БИС серии К588 на дополняющих МДП-транзисторах//Электронная промышленность, 1979, вып. 10. С. 36—38.

10. **Брук Б. И., Любович Л. А.** Некоторые вопросы высокочастотных связей в быстродействующих ЦВМ//ИТМ ВТ АН СССР. М.: 1968. С. 100.

11. **Васенков А. А., Коночкин Э. И., Малашевич Б. М.** Терминология в технике микропроцессорных интегральных схем и микроЭВМ//Микроэлектроника и полупроводниковые приборы/Под ред. А. А. Васенкова, Я. А. Федотова. М.: Сов. радио, 1979, вып. 4. С. 17—29.

12. **Весноватов М. Г., Домнин С. Б., Иванов Е. А.** Комплексные средства отладки микропроцессорных устройств//Электронная промышленность, 1986, вып. 9. С. 3—8.

13. **Глушков В. М.** Синтез цифровых автоматов. М.: Физматгиз, 1962. 467 с.

14. **Глушкова Г. Г., Иванов Е. А.** МикроЭВМ семейства «Электроника»// Микропроцессорные средства и системы, 1986, № 4. С. 7—11.

15. **ГОСТ** 15971—84. Системы обработки данных. Термины и определения.

16. **ГОСТ** 17021—75, с изменениями 1981 г. Микросхемы интегральные. Термины и определения.

17. **ГОСТ** 17467—79. Микросхемы интегральные. Основные размеры.

18. **ГОСТ** 19480—74, с изменениями 1981 и 1985 гг. Микросхемы интегральные. Термины, определения и буквенные обозначения электрических параметров.

19. **ГОСТ** 19781—83. Програмное обеспечение систем обработки данных. Термины и определения.

20. **Гришин Г. Г., Сухарев С. С.** MICROP — язык микропрограммирования для серии К589//Тр. НТО им. акад. А. Н. Крылова, № 32. С. 198.

21. **В. А. Гуляев.** Контроль ЭВМ. Киев: Наукова думка, 1977. 160 с.

22. **В. А. Гуляев.** Оптимальный синтез систем контроля цифровых управляющих машин//Теория точности и надежности кибернетических систем. Киев: изд-во ин-та кибернетики АН УССР, 1969.

23. **Гусев В. Н., Купцов В. В., Пантелеев С. В.** Интеллектуальный терминал на базе ДВК-1//Микропроцессорные средства и системы, 1986, № 1. С. 59—60.

24. **Дшхунян В. Л., Борщенко Ю. И., Отрохов Ю. Л.** Одноплатные микроЭВМ ряда «Электроника МС1201»//Микропроцессорные средства и системы, 1985, № 2. С. 8—13.

25. **Жинтелис Г. Б., Гринис В. А., Карияускас Э. К.** Система разработки программного обеспечения для секционируемых микропроцессоров//Автоматизация программирования аналогово-цифровых и микропроцессорных систем. М.: МДНТП им. Ф. Э. Дзержинского, 1982. С. 129—132.

26. **Забара С. С., Мильнер А. Д.** Проблемно-ориентированный комплекс автоматизированных рабочих мест для программирования МП-устройств. Киев: Общество «Знание», 1981.

27. **Заботкин Н. М., Загубин П. В., Каганов Ф. А.** Системы автоматизации и отладки программ и микропрограмм для микроЭВМ//Программное обеспечение АСУ. Секция III. Калинин, 1980. С. 122—125 (Тез. докл.).

28. **Злотник Е. М.** Секционированные микропроцессоры/Под ред. О. И. Семенова. Минск: Наука и техника, 1984. 191 с.

29. **Зубашич В. Ф.** и др. Микропроцессорный комплект БИС серии К580. Семейство микроЭВМ «Электроника К1»//Электронная промышленность, 1979, вып. 11. С. 19—22.

30. **Иванников А. Д.** Аппаратно-программные комплексы для отладки микропроцессорных устройств//Микроэлектроника и полупроводниковые приборы/Под ред. А. А. Васенкова и Я. А. Федотова. М.: Радио и связь, 1981, вып. 6. С. 62—82.

31. **Иванников А. Д., Старых В. А.** Проектирование микропроцессорных систем//Зарубежная электронная техника, 1980, вып. 11. С. 3—92.

32. **Иванников А. Д.** Организация кросс-системы разработки программного обеспечения микропроцессорных устройств//Электронная техника. Микроэлектроника. М.: ЦНИИ «Электроника», 1983, вып. 2. С. 37—44.

33. **Иванников А. Д.** Средства разработки и отладки микропрограммного обеспечения микропроцессорных систем//Программное обеспечение микропроцессорных устройств и микроЭВМ. М.: МДНТП им. Ф. Э. Дзержинского, 1984. С. 80—85.

34. **Интегральные** микросхемы: Справочник/Тарабрин Б. В., Лунин Л. Ф., Смирнов Ю. Н. и др. Под ред. Б. В. Тарабрина. М.: Радио и связь, 1984. 528 с.

35. **Каган Б. М.** Электронные вычислительные машины и системы: Учеб. пособие для ВУЗов. 2-е изд., перераб. и доп. М.: Энергоатомиздат, 1985. 552 с.

36. **Капрелянц И. Г., Рябов С. А., Филимонов А. Н.** Схемный эмулятор микропроцессорной БИС К1810ВМ86//Электронная промышленность, 1986, № 9. С. 17—18.

37. **Клингман Э.** Проектирование специализированных микропроцессорных систем/Пер. с англ. М.: Мир, 1979. 363 с.

38. **Кобылинский А. В., Москалевский А. И., Темченко В. А.** Однокристальный высокопроизводительный 16-разрядный микропроцессор КМ1810ВМ86//Микропроцессорные средства и системы, 1986, № 1. С. 28—33.

39. **Кобылинский А. В., Темченко В. А.** Система команд микропроцессора КМ1810ВМ86//Микропроцессорные средства и системы, 1986, № 3. С. 30—35.

40. **Кузнецов П. И., Пчелинцев Л. А., Гайденко В. С.** Контроль и поиск неисправностей в сложных системах. М.: Сов. радио, 1969. 178 с.

41. **Мальцев П. П., Харьков В. В.** Особенности структуры центральных процессоров на основе БИС серии К588, К589, К1804//Микропроцессорные средства и системы, 1985, № 2. С. 5—53.

42. **Мик Дж., Брик Дж.** Проектирование микропроцессорных устройств с разрядно-модульной организацией: В 2-х книгах/Пер. с англ. М.: Мир, 1984.

43. **Мик Дж., Скопмейф Г.** Микропроцессор для отладки микропрограммируемых машин на основе процессорных секций//Электроника, 1977. Т. 50, № 19. С. 51—56.
44. **Микропроцессорные** БИС и микроЭВМ: Построение и применение/ Васенков А. А., Воробьев Н. М., Дшхунян В. Л. и др. Под ред. А. А. Васенкова. М.: Сов. радио, 1980. 280 с.
45. **Микропроцессорные** комплекты интегральных схем: состав и структура: Справочник/ Борисов В. С., Васенков, А. А., Малашевич Б. М. и др. Под ред. А. А. Васенкова, В. А. Шахнова. М.: Радио и связь, 1982. 192 с.
46. **Мильнер А. Д.** и др. Принципы построения проблемно-ориентированного комплекса для программирования МП-устройств и система отладки на его базе// Машинное моделирование. М.: МДНТП им. Ф. Э. Дзержинского, 1980. С. 83—91.
47. **Михайлов С. А.** Система управления электроприводами судовых грузовых комплексов//Микропроцессорные средства и системы, 1985, № 4. С. 83—86.
48. **Мошков А. А., Овечкин Ю. А., Ольшанский О. В.** Контроллеры на базе микропроцессорных БИС//Судостроение за рубежом, 1982, № 6. С. 40—51.
49. **Наумов Ю. Е., Аваев Н. А., Бедрековский М. А.** Помехоустойчивость устройств на интегральных логических схемах. М.: Сов. радио, 1975. 215 с.
50. **Однокристальные** микропроцессоры комплекта БИС серии К1801/Дшхунян В. Л., Борщенко Ю. И., Науменков В. Р. и др.//Микропроцессорные средства и системы, 1984, № 4. С. 12—18.
51. **Однокристальный** микропроцессор КМ1801ВМ3/Волков Р. И., Горский В. П., Дшхунян В. Л. и др.//Микропроцессорные средства и системы, 1986, № 4. С. 37—41.
52. **Одноплатное** процессорное устройство в Евростандарте на БИС серии К1801/Бабиченко С. И., Власкин А. Б., Гриценко А. Ф. и др.//Микропроцессорные средства и системы, 1984, № 4. С. 25—27.
53. **Основы** построения технических средств ЕС ЭВМ на интегральных микросхемах/Саморуков В. В., Микитин В. М., Павлычев В. А. и др. Под ред. Б. Н. Файзулаева. М.: Радио и связь, 1981. 288 с.
54. **Пекелис В. Г., Симхес В. Я.** Паразитные наводки в быстродействующих ЭЦВМ. Минск: Наука и техника, 1967. 108 с.
55. **Попов А. А., Хохлов М. М., Глухман В. Л.** Диалоговые вычислительные комплексы «Электроника НЦ-80-20»//Микропроцессорные средства и системы, 1984, № 4. С. 61—64.
56. **Проектирование** цифровых систем на комплектах микропрограммируемых БИС/Булгаков С. С., Мещеряков В. М., Новоселов В. В. и др. Под ред. В. Г. Колесникова. М.: Радио и связь, 1984. 240 с.
57. **Сыпчук П. П.** Инструментальные средства разработки программного обеспечения микропроцессорных устройств и систем//Программное обеспечение микропроцессорных устройств и микроЭВМ. М.: МДНТП им. Ф. Э. Дзержинского, 1984. С. 41—45.
58. Унифицированные блоки питания для ЭВМ/Губанов В. Н.//Вопросы радиоэлектроники. Сер. ЭВТ, 1971, вып. 12. С. 118—127.
59. **Филлипычев С. А., Майдаковский И. В., Борщенко Ю. И.** Применение однокристального микропроцессора К1801ВМ1 в автономных системах сбора и обработки информации//Микропроцессорные средства и системы, 1985, № 1. С. 51—57.
60. **Ханмамедов С., Денисов В., Дульнев И.** Не разбирать вспомогательные механизмы//Морской флот, 1983, № 4. С. 49.
61. **Юрымов Н.** На танкере «Поль Робсон»//Морской флот, 1983, № 4. С. 48.
62. **Corconar P.** The SUMA microprogramming system//Microprocessing and Microprogramming, 1981, N 7. P. 37—45.
63. **Microprogramming** The Series 3000. Intel Corporation, 1975.
64. **Tamira E., Tokoro M.** Hierarchical Micropragraming. Generating System// Micro, 1979, 12.

ОГЛАВЛЕНИЕ